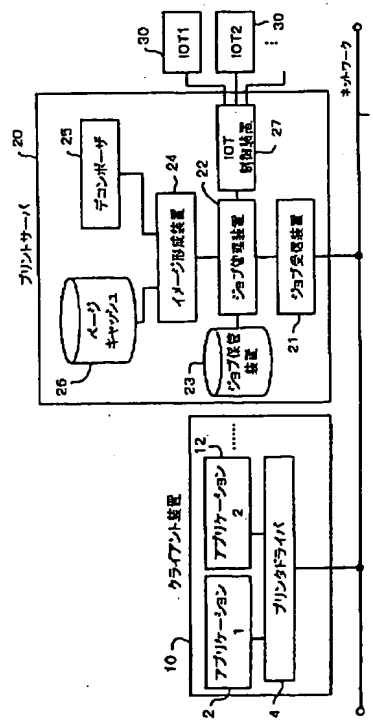


(11)特許出願公開番号
特開2000-207150
(P2000-207150A)



【特許請求の範囲】

【請求項1】 少なくとも1つの文書構成要素の印刷指示データを含む印刷ジョブを作成するクライアント装置と、

印刷ジョブを解釈して各文書構成要素の印刷可能データを生成し、印刷装置に供給して印刷させるプリントサーバと、

を含み、

前記クライアント装置は、新規生成又は変更した文書構成要素に対し、少なくとも前記印刷システム内で一意的な識別子を付与する手段と、付与された識別子と文書構成要素との対応関係の情報を印刷ジョブに組み込む手段と、を有し、

前記プリントサーバは、各文書構成要素の印刷指示データから生成した印刷可能データに対し、当該文書構成要素に付与された識別子に対応づけて記憶する記憶手段を有し、

前記識別子を用いて前記プリントサーバに記憶された印刷可能データを再利用可能としたことを特徴とする印刷システム。

【請求項2】 前記クライアント装置における識別子と文書構成要素との対応関係の情報を印刷ジョブに組み込む手段は、前記識別子を、対応する文書構成要素を記述した印刷指示データに対するコメント文の形で印刷ジョブ内に組み込むことを特徴とする請求項1に記載の印刷システム。

【請求項3】 前記プリントサーバは、印刷ジョブのコメント文を解析して各文書構成要素に付与された識別子を検出する手段と、検出した識別子に対応する印刷可能データが前記記憶手段に記憶されているか否かを判定する手段と、記憶されていると判定された場合、当該文書構成要素の印刷指示データから印刷可能データを生成する代わりに、前記検出した識別子に対応する印刷可能データを前記記憶手段から読み出し、その印刷指示データから生成した印刷可能データとして利用する手段と、を有することを特徴とする請求項2記載の印刷システム。

【請求項4】 前記クライアント装置は、前記プリントサーバに対し、印刷ジョブを構成する各文書構成要素の印刷可能データが前記記憶手段に記憶されているか否かを、各文書構成要素の識別子を用いて問い合わせる手段と、

この問い合わせの結果、前記記憶手段に印刷可能データが記憶されているとの回答を受けた文書構成要素については、当該文書構成要素を記述した印刷指示データを作成する代わりに、前記記憶手段に記憶された該文書構成要素の識別子に対応する印刷可能データを読み出して使用する旨の記述を生成し、印刷ジョブに組み込む手段と、を有することを特徴とする請求項1記載の印刷システム。

ム。

【請求項5】 前記印刷システムは複数の前記プリントサーバを有し、前記各プリントサーバは、クライアント装置から受けた印刷ジョブに含まれる文書構成要素の識別子に対応する印刷可能データが自己の記憶手段にない場合は、他のプリントサーバに対して当該識別子に対応する印刷可能データを記憶しているか否かを問い合わせ、当該識別子に対応する印刷可能データを記憶しているプリントサーバがあった場合は、そのプリントサーバから当該印刷可能データの供給を受けることを特徴とする請求項1から請求項4のいずれかに記載の印刷システム。

【請求項6】 前記文書構成要素はページであることを特徴とする請求項1から請求項5のいずれかに記載の印刷システム。

【請求項7】 文書を構成する各文書構成要素ごとに固有の識別子を付与してその更新履歴を管理するアプリケーションの作成した文書データから、その文書の印刷イメージを記述した所定のページ記述言語の印刷ジョブを生成するプリンタドライバであって、文書データから、当該文書の各文書構成要素の識別子を抽出する手段と、

前記各文書構成要素を表すページ記述言語の記述を生成する手段と、

前記各文書構成要素を表すページ記述言語の記述に対し、当該文書構成要素の識別子の情報を当該ページ記述言語に規定されたコメント文の形式で付加して印刷ジョブを生成する手段と、

を有するプリンタドライバ。

【請求項8】 ページ記述言語で記述された印刷ジョブを解釈して印刷可能データを生成する解釈手段を有し、生成した印刷可能データを印刷装置に供給して印刷させるプリントサーバであって、

前記解釈手段で生成した印刷可能データを記憶する記憶手段であって、文書の各文書構成要素ごとに、その文書構成要素の印刷可能データとその文書構成要素の識別子とを対応づけて記憶する記憶手段と、印刷ジョブを解析して、当該印刷ジョブの表す文書の各文書構成要素の識別子を検出する手段と、

検出した識別子に対応する印刷可能データが前記記憶手段に記憶されているか否かを判定する手段と、

記憶されていないと判定された場合は、前記検出した識別子に対応する文書構成要素の印刷指示データを前記解釈手段に解釈させて当該文書構成要素の印刷可能データを生成し、記憶されていると判定された場合は、前記検出した識別子に対応する印刷可能データを前記記憶手段から読み出して出力する手段と、

前記解釈手段で印刷可能データを生成した文書構成要素について、その文書構成要素の識別子と印刷可能データを前記記憶手段に記憶させる手段と、

を有するプリントサーバ。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、クライアント装置で生成した印刷指示データをプリントサーバで解釈して印刷可能データを生成し、これを印刷装置に供給して印刷させる印刷システムにおいて、再印刷の処理負荷を軽減するための技術に関する。

【0002】

【従来の技術】再印刷を効率よく行うために、プリントサーバとクライアント装置の間、もしくはプリントサーバの中に、文書を保存・管理する文書管理装置が設けられることがある。このようなシステムでは、ある文書の再印刷の必要が生じた場合、文書管理装置から、以前の印刷時に保存しておいた同文書のデータを取り出して印刷することで、クライアント装置で再び同じ文書の印刷指示データを生成してプリントサーバへ送信する手間を省き、再印刷の効率を向上させている。

【0003】さて、文書編集から印刷までの一連の処理を考えた場合、一つの文書がそれぞれの処理段階において適切な様々なファイルフォーマットに変換されて扱われることはよく知られている。各フォーマットにはそれぞれ長所短所があり、文書管理装置で保管する文書のファイルフォーマットをどうするかは、システムの効率を考える上で重要なポイントとなる。

【0004】文書編集アプリケーションの（もしくはそれに近い）ファイルフォーマットのは、文書の編集には適しているが、アプリケーション固有のフォーマットである場合がほとんどで汎用性の点で問題がある。また、ファイルフォーマットの仕様自体が公開されていない場合もあり、このような場合は文書保管装置の構築がそもそも困難となる。また、文書編集アプリケーションのファイルフォーマットを用いた場合、再印刷の指示から完了までに時間を要したり、最初の印刷時と異なる印刷装置を出力先に指定した場合などにはフォントの不一致などから同じ印刷結果が得られない可能性もあり、このフォーマットは再印刷用の保管データのフォーマットに適しているとは言い難い。

【0005】一方、プリンタの内部で最終的に使用されるファイルフォーマット（例えばラスタイメージ）は、通常、そのプリンタ（あるいはプリンタベンダ）独自のものになり、他のプリンタなどで汎用的に利用することはできない。プリントサーバに、さまざまなベンダが提供するさまざまなプリンタを接続するシステム構成を考えた場合、そのようなフォーマットは再印刷用の文書保管フォーマットには適さない。

【0006】このような理由から、再印刷を目的とした文書管理装置では、保管する文書のフォーマットとしてPDL（ページ記述言語）を用いることが多い。PDLは、ページを記述する言語であり、一般に、クライアン

ト装置からプリントサーバへの文書データ転送時のフォーマットとして用いられている。PDLは、プリンタなどの個々のデバイスの機種に依存しないページ記述を目指したものであり、たとえば業界標準的なPostScript（米国アドビシステムズ社の商標）など、多くのベンダのプリントサーバでサポートされているものもある。PDLを文書の保管用フォーマットとして用いた場合、さまざまなプリンタで汎用的に利用できる上、再印刷結果の同一性をかなりの程度保証することができ、再印刷時の印刷速度も文書編集アプリケーションのフォーマットに比べ改善される。

【0007】しかしながら、PDLは、プリンタで処理可能なラスタイメージに変換するのにある程度の時間を要する。特にカラーイメージを含んだ文書は、周知のようにPDLからラスタイメージへの展開にかなり長い時間を要する。このため、PDL形式で文書を保管するシステムでは、再印刷速度の改善が要望されている。

【0008】この要望に応えるシステムとして、文書をPDLファイルの形で保管するだけでなく、特定のプリンタでの印刷に特化したフォーマット（PRF（プリントレディフォーマット）と呼ぶ）のファイルも合わせて保管するシステムが存在する。PRFとは、例えば解像度やイメージ主走査方向などが該プリンタが最も効率的に働くように設定されたフォーマット（一般にはラスタの一種）であり、通常は、PDLファイルを展開処理した結果作成される。このようなシステムでは、文書の再印刷を前回印刷時と同じプリンタで行う場合は、PDLと対応づけて保管した当該文書のPRFファイルを利用することにより、高速に再印刷処理を行うことができる。

【0009】

【発明が解決しようとする課題】しかしながら、このような従来システムは、PDLファイルを管理の単位として文書管理を行っており、アプリケーションレベルで若干の文書の修正が行われるような、なかり頻繁に発生する事象が起こった場合には非効率な事態が生じる。例えば、アプリケーション上で印刷後に誤字を発見して修正した場合などには、たった1文字の修正でも、その文書全体のPDLファイルを作成し直すことになり、その場合PDLファイル自体が変更されたので、以前作成しておいた修正前の当該文書のPRFファイルは使用できず、作成し直されたPDLファイルに対し再び処理の遅い展開処理を行う必要があった。

【0010】つまり、上記従来のPDLベースの文書管理は、印刷を目前にした修正のない完成された文書を管理するには適しているが、修正が頻繁に行われる文書を取り扱う環境では、せっかく以前に作ったPRFファイルが利用できず、再印刷の高速化等のメリットを得ることができなかった。

【0011】本発明は、以上の問題を解決するためにな

されたものであり、一度印刷した文書のデータを保管して再印刷に利用するシステムにおいて、文書修正が頻繁に行われる環境でも、以前に印刷した文書のデータを有効利用し、修正後の再印刷の処理速度を改善することを目的とする。

【0012】

【課題を解決するための手段】上記目的を達成するため、本発明に係る印刷システムは、少なくとも1つの文書構成要素の印刷指示データを含む印刷ジョブを作成するクライアント装置と、印刷ジョブを解釈して各文書構成要素の印刷可能データを生成し、印刷装置に供給して印刷させるプリントサーバと、を含み、前記クライアント装置は、新規生成又は変更した文書構成要素に対し、少なくとも前記印刷システム内で一意的な識別子を付与する手段と、付与された識別子と文書構成要素との対応関係の情報を印刷ジョブに組み込む手段と、を有し、前記プリントサーバは、各文書構成要素の印刷指示データから生成した印刷可能データに対し、当該文書構成要素に付与された識別子に対応づけて記憶する記憶手段を有し、前記識別子を用いて前記プリントサーバに記憶された印刷可能データを再利用可能としたことを特徴とする。

【0013】この構成では、クライアント装置側で印刷ジョブに含まれる個々の文書構成要素ごとに一意的な識別子を付与し、プリントサーバ側で各文書構成要素の印刷可能データをこの識別子と対応づけて保存することにより、印刷可能データをPDFファイルの単位ではなく、文書構成要素の単位で再利用することが可能となった。文書構成要素は、例えばページであってもよく、また個々の文字列や図形、イメージなどの描画オブジェクトであってもよい。この構成では、1つの文書の中でも、修正された文書構成要素には修正前と異なる識別子が付与され、修正されなかった文書構成要素には元と同じ識別子が付与される。したがって、一度印刷した文書に対し修正を行った場合、修正していない文書構成要素については、前に印刷に用いた印刷可能データを再利用して印刷することができ、印刷指示データを印刷可能データに展開する処理を省くことができる。

【0014】本発明の好適な態様では、クライアント装置は、識別子を、対応する文書構成要素を記述した印刷指示データに対するコメント文の形で印刷ジョブ内に組み込む。コメント文は、プリントサーバの有する印刷可能データの生成手段（いわゆるインタプリタ）には影響を与えない形式である。プリントサーバに、インタプリタの他に、印刷ジョブ中のコメント文を解析して識別子を検出する手段を設ければ、既に記憶手段に印刷可能データが記憶済みの文書構成要素については、検出した識別子に基づき対応する印刷可能データを検索して利用することができる。また、この態様では、印刷ジョブには各文書構成要素を記述した印刷指示データが含まれてい

るので、本発明に対応しているプリントサーバで処理しても対応していないプリントサーバで処理しても、基本的に同じ印刷結果が得られる。したがって、この態様は、ネットワーク上のすべてのプリントサーバが本発明の識別子を利用した仕組みを持っていない場合でも矛盾なく使用することができ、多数のプリントサーバが接続されたネットワーク環境にも容易に導入することができる。

【0015】また、本発明の別の好適な態様では、クライアント装置は、プリントサーバに対し、印刷ジョブを構成する各文書構成要素の印刷可能データが記憶手段に記憶されているか否かを、各文書構成要素の識別子を用いて問い合わせ、この問合せの結果、記憶手段に印刷可能データが記憶されているとの回答を受けた文書構成要素については、当該文書構成要素を記述した印刷指示データを作成する代わりに、記憶手段に記憶された該文書構成要素の識別子に対応する印刷可能データを読み出して使用する旨を指示する印刷指示データを生成し、印刷ジョブに組み込む。

【0016】この態様によれば、印刷ジョブのデータ量を少なくすることができ、その作成や伝送に要する時間を削減することができる。

【0017】また、本発明の別の好適な態様では、複数の前記プリントサーバを有する印刷システムにおいて、各プリントサーバは、クライアント装置から受けた印刷ジョブに含まれる文書構成要素の識別子に対応する印刷可能データが自己の記憶手段にない場合は、他のプリントサーバに対して当該識別子に対応する印刷可能データを記憶しているか否かを問い合わせ、当該識別子に対応する印刷可能データを記憶しているプリントサーバがあった場合は、そのプリントサーバから当該印刷可能データの供給を受ける。

【0018】この態様によれば、複数のプリントサーバを有するシステムにおいても、以前に生成した印刷可能イメージをできるだけ有効に活用することにより、印刷ジョブの解釈に要する時間を低減し、再印刷の処理時間を短縮することができる。

【0019】また、本発明は、文書を構成する各文書構成要素ごとに固有の識別子を付与してその更新履歴を管理するアプリケーションの作成した文書データから、その文書の印刷イメージを記述した所定のページ記述言語の印刷ジョブを生成するプリンタドライバであって、文書データから、当該文書の各文書構成要素の識別子を抽出する手段と、前記各文書構成要素を表すページ記述言語の記述を生成する手段と、前記各文書構成要素を表すページ記述言語の記述に対し、当該文書構成要素の識別子の情報を当該ページ記述言語に規定されたコメント文の形式で付加して印刷ジョブを生成する手段とを有するプリンタドライバを提供するものである。

【0020】また、本発明は、ページ記述言語で記述さ

れた印刷ジョブを解釈して印刷可能データを生成する解釈手段を有し、生成した印刷可能データを印刷装置に供給して印刷させるプリントサーバであって、前記解釈手段で生成した印刷可能データを記憶する記憶手段であって、文書の各文書構成要素ごとに、その文書構成要素の印刷可能データとその文書構成要素の識別子とを対応づけて記憶する記憶手段と、印刷ジョブを解析して、当該印刷ジョブの表す文書の各文書構成要素の識別子を検出する手段と、検出した識別子に対応する印刷可能データが前記記憶手段に記憶されているか否かを判定する手段と、記憶されていないと判定された場合は、前記検出した識別子に対応する文書構成要素の印刷指示データを前記解釈手段に解釈させて当該文書構成要素の印刷可能データを生成し、記憶されていると判定された場合は、前記検出した識別子に対応する印刷可能データを前記記憶手段から読み出して出力する手段と、前記解釈手段で印刷可能データを生成した文書構成要素について、その文書構成要素の識別子と印刷可能データを前記記憶手段に記憶させる手段とを有するプリントサーバを提供するものである。

【0021】

【発明の実施の形態】以下、本発明の実施の形態（以下実施形態という）について、図面に基づいて説明する。

【0022】図1は、本発明に係る印刷システムの概略構成を示す機能ブロック図である。印刷システムは、ネットワーク40を介して相互に接続されたクライアント装置10及びプリントサーバ20と、プリントサーバ20に専用ケーブル等で接続されたIOT（Image Output Terminal）30から構成される。

【0023】クライアント装置10は、コンピュータであり、各種のアプリケーション12を搭載する。それらアプリケーション12で作成された文書に対し、ユーザから印刷指示が行われると、プリンタドライバ14は、アプリケーション12のデータ形式の文書をページ記述言語（PDL）のデータ形式に変換して印刷ジョブを生成し、プリントサーバ20に送信する。プリントサーバ20は、この印刷ジョブを解釈してIOT30に扱えるラスティメージなどの印刷可能データに変換する。IOT30は、プリントエンジンとも呼ばれ、プリントサーバ20から供給される印刷可能データに基づき、用紙上に文書の可視イメージを形成する。IOT30は、1つのプリントサーバ20に対して複数接続することができる。また、クライアント装置10及びプリントサーバ20も、ネットワーク40に複数接続することができる。

【0024】以下、クライアント装置10とプリントサーバ20の内部構成を更に詳しく説明する。

【0025】【クライアント装置】

（1）アプリケーション12

クライアント装置10のアプリケーション12は、文書の各ページに固有な識別番号（ページIDと呼ぶ）を付

与し、各ページの修正履歴を管理する機能を持つ。各ページには、新規生成されたときにページIDが付与されると共に、保存されていた状態からそのページが変更されたら、そのページには新たなページIDが付与される。アプリケーション12が作成する文書データには、各ページのページIDが含まれる。なお、ページIDは、少なくとも当該印刷システムの内部で一意的になるように付与する。

【0026】ページID付与には、例えばMicrosoft社が利用しているGUID（Global Unique ID）を用いることができる。これは、各装置ごとにそれぞれ固有のIDを付与しておき、各装置が生成したデータに対しその装置内での一意的なIDを付与すると共に、そのIDにその装置自体のIDを冠することにより、そのデータのIDを作成するアルゴリズムであり、ほぼ確実にグローバルに一意的な128ビットのIDを作成することができる。GUIDを用いたアプリケーションとして、例えばMicrosoft社のワードプロセッサアプリケーションであるMS Word（商品名）があり、これを本実施形態におけるアプリケーション12として用いることもできる。なお、ページID付与のアルゴリズムはこれに限られるものではなく、当該印刷システム内で一意性を保証できるものであればどのようなものでもよい。

【0027】（2）プリンタドライバ14

プリンタドライバ14は、アプリケーション12と協業して印刷ジョブ（以下単にジョブという）を作成する。ジョブは、例えば、文書の各ページの印刷イメージをPDLで記述したPDLファイルと、部数やフィニッシング処理（ステーブル留めなどの後処理）などの印刷処理属性の指定を記述したジョブチケットファイルから構成される。なお、ジョブチケットファイルは本発明にとって本質的なものではなく、ジョブチケットファイルを含まない印刷ジョブを生成するシステムにも本発明が適用可能であることは、以降の説明から明確に理解されよう。

【0028】印刷ジョブのPDLファイルの作成に当たり、プリンタドライバ14は、アプリケーション12の文書データから、その文書の各ページの印刷イメージを表すPDL記述を作成すると共に、文書データからそれら各ページのページIDを検出し、PDLの記述の中に組み込む。ページIDのPDLファイルへの組み込みは、例えばコメント文の形式を用いることにより行うことができる。PDLとしてPostScript（米国アドビシステムズ社の商標）を用いた場合、例えば図2に示すような記述となる。この例では、「%%PageID:」で始まる行がページIDを表すコメント文であり、{}内値がページIDである。なお、PostScriptでは記号「%」で始まる行はコメントと認識され、インタプリタはこれを解釈しない。「%%Page: 77」なる記述は、ある1ページ（この例では77ページ）のイメー

ジを記述したPDL記述の始まりを示すコメント文であり、次行のページIDはこのページに付与されたものである。このように、この例では、ページIDの記述は、当該ページのPDL記述の先頭部分に埋め込まれる。

【0029】ページIDをコメント文としてPDL記述中に組み込むというこの方式によれば、プリントサーバ20側では、ページIDの記述を検出、解析するために、コメント文を解析してページIDを抽出する機構(プログラム)をPDLのインタプリタとは別に設ければよく、PDLのインタプリタ自体には何ら変更を加える必要がない。また、コメント文はPDLのインタプリタでは無視され、インタプリタには何の影響も与えないので、この方式には、ページIDのコメント文を解析する機構を持たないプリントサーバに対しても互換性を失わないという利点もある。

【0030】この方式では、プリンタドライバ14はアプリケーション12と協業して、ユーザーに指示された範囲のすべてのページを表すPDL記述を作成し、そこに各ページのページIDをコメント文として埋め込む。本実施形態では、既にプリントサーバ20で印刷されたことのあるページは、その時作成した印刷可能データを再利用する(詳細は後述)ので、すべてのページを含むPDLファイルを作成することは一見無駄に見えるかもしれないが、これは、ページIDを認識できないシステムでも問題なく印刷できるようにするためのものである。すなわち、この方式により、ネットワーク40上にページIDを認識できる本実施形態のプリントサーバ20と、ページIDを認識できないプリントサーバとが混在する場合でも、プリンタドライバ14はそのような混在を意識せずに全く同じPDLファイルを作成することができる。

【0031】このようなページIDが埋め込まれたジョブによれば、プリントサーバ20は、各ページが既に一度印刷したことのあるものか、それとも初めて受け取ったものかを判別することができる。例えば、ある文書1のバージョンAのジョブがすでにプリントサーバ20に送信され、印刷も完了している状態で、クライアント装置10側でページX及びYのみが変更されたその文書1のバージョンBが作成され、プリントサーバ20に送られた場合、プリントサーバ20では、各ページのページIDを調べることにより、ページX及びYのみが変更されたことが理解できる。他のページのページIDは前のバージョンのときと同じであるが、ページX及びYだけ前のバージョンとは異なる未知のページIDを持っているからである。

【0032】このようにしてプリンタドライバ14で作成されたPDLファイルは、別途作成されたジョブチケットファイルと共に、ジョブとしてネットワーク40を介しプリントサーバ20に送信される。

【0033】[プリントサーバ]

(1) ジョブ受信装置21

プリントサーバ20では、ジョブ受信装置21がネットワーク40からジョブ受信する。受信したジョブはジョブ管理装置22に送られ、管理される。

【0034】(2) ジョブ管理装置22

ジョブ管理装置22は、受信したジョブを保管し、印刷の流れ全体を統括制御する。ジョブ管理装置22の処理手順を図3を参照して説明する。

【0035】ジョブ管理装置22は、ジョブ受信装置21からジョブを受け取ると(S10)、まず受信したジョブファイル、すなわちPDLファイルとジョブチケットファイルをジョブ保管装置23に送り、それらファイルの保管を依頼する(S12)。ジョブ保管装置23は、その依頼に応じてそれらファイルを保管する(S22。詳細は後述)。また、ジョブ管理装置22は、それら受信ジョブの処理順序を管理する。すなわちジョブ管理装置22は、受信したジョブを印刷待ちのキューに登録し、スケジューリングする。そして、ジョブを印刷する順番が来ると、そのジョブをイメージ形成装置24に送り、ページイメージの作成を依頼する(S14)。この依頼の後、ジョブ管理装置22は、イメージ形成処理の完了待ちの状態となる(S16)。イメージ形成装置24は、この依頼に応じて、デコンポーザ25とページキャッシュ26を制御して、そのジョブの各ページのページイメージを作成し、作成したページイメージをページキャッシュ26に保存すると共に、それらページキャッシュ26に保存した各ページイメージへのアクセス情報(各ページイメージのファイル名など)を示したファイル(構成ページファイルと呼ぶ)を作成し、ジョブ管理装置22に渡す。ジョブ管理装置22は、この構成ページファイルをジョブ保管装置23に送り、元のジョブのファイル(PDLファイルやジョブチケットファイル)との関連付け処理を依頼する(S18)。この依頼に応じ、ジョブ保管装置23は、構成ページファイルをジョブファイルに対応づけて保管する(S26。詳細は後述)。そして、ジョブ管理装置22は、その構成ページファイルをジョブチケットの情報とともにIOT制御装置27に送り、印刷処理を依頼する(S20)。IOT制御装置27は、この依頼に応じ、構成ページファイルを参照して各ページのページイメージをページキャッシュ26から取り出し、IOT30に渡して印刷を行わせる(S28)。

【0036】(3) ジョブ保管装置23

ジョブ保管装置23は、各ジョブの情報を印刷や再印刷のために保管する、一種のデータベースである。ここに保管される情報には、クライアント装置10から受信したジョブファイル(すなわちPDLファイル及びジョブチケットファイル)や、ページキャッシュ26に保管されている各ページイメージとジョブとの関連付け情報である構成ページファイルが含まれる。ジョブ保管装置2

3にはジョブに関する一連の情報が保存されているため、プリントサーバ20のユーザインタフェース又はクライアント装置10からジョブ保管装置23にアクセスし、そこでジョブを指定することにより、そのジョブの再印刷をジョブ管理装置22に指示することができる。再印刷の場合、当該ジョブの各ページのページイメージはすでにページキャッシュ26に保存されているため、あらためてPDLファイルを解釈してイメージ作成を行う必要はない。

【0037】(4) イメージ形成装置24

イメージ形成装置24は、ジョブの文書を、IOT30で取扱可能な印刷可能データに変換する装置である。イメージ形成装置24は、ジョブ管理装置22からジョブのイメージ形成処理の依頼を受け、このとき同時に受け取ったPDLファイル及びジョブチケットファイルに従って、各ページの印刷可能データであるページイメージを作成する。

【0038】ここで注意すべきは、PDL解釈及び描画処理という実際のイメージ作成のための処理を行うのは、デコンポーザ25であるということである。イメージ作成という観点で言えば、イメージ形成装置24は、デコンポーザ25に対してイメージ作成を依頼し、イメージ作成処理全体を管理する上位装置として機能する。このような管理手段としてのイメージ形成装置24を設けるのは、ページキャッシュ26を利用したイメージ形成処理の効率化を実現するためである。

【0039】本実施形態では、この処理は、デコンポーザ25からのページキャッシュ問合せに応じて行われる。イメージ形成装置24は、PDLファイルに組み込まれたページIDを利用して、各ページのページイメージがページキャッシュ26内に既に格納済みかどうかを調べる。そして、イメージ形成装置24は、ジョブの各ページのページイメージがページキャッシュ26に保存されていないと判明した場合は、デコンポーザ25にそのページのページイメージを作成させ、ページキャッシュ26に既に保存されていると判明した場合は、デコンポーザ25にそのページのイメージの作成を省略させ、ページキャッシュ26中のそのページイメージを利用する。このような処理を各ページについて繰り返すことにより、ジョブの全ページのページイメージが、デコンポーザ25又はページキャッシュ26のいずれかから取得できる。

【0040】ジョブの全ページのページイメージが作成できると、イメージ形成装置24は、ジョブ管理装置22に対し、イメージ作成依頼の応答として、ジョブを構成する各ページのページイメージの情報を返す。本実施形態では、デコンポーザ25で作成されたページイメージはすべてページキャッシュ26に保存し、ジョブ管理装置22へは、ページキャッシュ26内のそれら各ページのイメージへアクセスするために必要な情報、すなわ

ち構成ページファイルを渡す。

【0041】この一連の処理の手順については後の(7)「イメージ形成処理の手順」で詳述する。

【0042】(5) デコンポーザ25

デコンポーザ25は、ジョブのPDLファイルを解釈して各ページのラスタイメージを作成し(この解釈、イメージ作成処理を合わせて「デコンボーズ処理」と呼ぶ)、イメージ形成装置24に返送する。デコンポーザ25は、PDLの記述を解釈するインタプリタに加え、コメント文を解析してページIDを検出し、そのIDに対応するページイメージが既にページキャッシュ26に保存されているかどうかを問い合わせるための機構を有する。デコンポーザ25は、PDLファイルの記述を先頭から順に調べていき、ページIDの検出すると、そのページIDに対応したページイメージファイルがページキャッシュ26に保存されているかどうか、イメージ形成装置24に対して問合せを行う。そして、この問合せにより、ページキャッシュ26内にそのページIDのページイメージが既に保存されていると分かった場合は、そのページについてのデコンボーズ処理を省略する。なお、デコンポーザ25の詳細な機能については、後の(7)で詳述する。

【0043】(6) ページキャッシュ26

ページキャッシュ26は、デコンポーザ25でのデコンボーズ処理によりPDLファイルから作成された各ページのページイメージを、ファイルとして保管し、管理する。ここで、ページイメージのファイルには、当該ページのページIDの情報が関連づけられる。例えば、ページID自体を当該ページイメージのファイル名とすることが好適である。

【0044】ページIDは一意的なものであるため、各ページイメージのファイルは、特にジョブごとに管理する必要はなく、すべてのジョブのすべてのページのイメージを同じレベルで均質に管理することができる。例えば、ファイルシステム上の1つのディレクトリをページキャッシュ26とし、すべてのページイメージファイルを管理することができる。

【0045】ページイメージのファイル名にページIDを用いた場合、{5a61f8f3-cdel-11cf-9113-00aa00425c62}というページIDを持つページのページイメージファイルは、{5a61f8f3-cdel-11cf-9113-00aa00425c62}というファイル名で保管する。このようにページキャッシュ中のページイメージファイルのファイル名としてページIDを用いると、デコンポーザ25からの問合せに応じて特定のページIDを持つページイメージファイルを検索する際、ページキャッシュ26を構成するディレクトリの中のファイル名を検索するだけでよい。

【0046】なお、この代わりに、ページイメージファイルをジョブごとに整理して管理する方式も考えられる。例えば、ページキャッシュ26のディレクトリ中に

10

20

30

40

50

ジョブごとにサブディレクトリを作成し、そのサブディレクトリ下にそのジョブのページイメージファイルを保存するなどである。この方式の方が検索性がよく、ジョブ単位での再印刷指示を受けた場合などにはこちらの方が高速に対応できる。

【0047】ページキャッシュ26は、典型的にはハードディスク装置のような大容量記憶装置を用いて実現される。ページキャッシュ26は、このような記憶装置の容量に依存し、有限の大きさであるため、ページイメージファイルを無限に保存することはできない。このため、不要なページイメージファイルを適宜削除する機能を持つ必要がある。この機能は、一般的なキャッシュメモリにおいて公知の様々な手法を用いて実現できる。例えば、ページイメージの総量が予め定めた上限値を超えたときに、最も古く作成されたページイメージから削除したり、最も長い間参照されないページイメージから削除したりするという方式である。また、ページイメージの利用頻度の統計データを順次作成し、この統計データをもとに削除するページイメージを決めてもよい。また、ページイメージの総量が上限値を超えたときに、サーバ管理者に警告を出し、管理者に削除させるような仕組みも可能である。

【0048】(7) イメージ形成処理の手順

ここで、図5を参照して、これまでに説明したイメージ形成装置24、デコンポーザ25及びページキャッシュ26によるイメージ形成処理の流れを説明する。

【0049】<処理の開始>イメージ形成処理は、イメージ形成装置24がジョブ管理装置22からジョブの処理依頼を受信したときから始まる。イメージ形成装置24は、ジョブ管理装置22からジョブファイル(PDLファイル及びジョブチケットファイル)を受信してその処理依頼を受けると(S30)、まずページIDリストの初期化を行う(S32)。このリストは、構成ページファイルの作成のために用いるものであり、以降の処理で、イメージ形成した各ページのページIDが順次登録されることになる。そして、イメージ形成装置24は、ジョブチケットファイルからイメージ形成に必要な情報を取り出し、この情報をPDLファイルと共にデコンポーザ25に渡し、ページイメージの作成を依頼する(S34)。

【0050】<デコンポーザの処理>デコンポーザ25は、それらの情報を受け取ると(S50)、デコンポーザ処理を開始する。デコンポーザ25は、まずページイメージの形成に用いる各種の状態情報や描画メモリを初期化し(S52)、PDLファイルを先頭から順に読み込み、処理していく。このとき、まずページIDの記述の有無を検査する(S54)。本実施形態では、前述のように、各ページのページ内容のPDL記述の前に、コメント文の形で当該ページのページ番号及びページIDが組み込まれている。S54ではこのページIDの記述

の有無を調べる。ページID記述の有無に応じて以降の処理が変わる。

【0051】<ページID記述がなかった場合>ページIDの記述が検出できなかった場合、すなわちPDLファイルにおいてページIDの記述なしでページ内容の記述が始まっている場合は、デコンポーザ25は、当該ページに対するページIDを作成し(S58)、そのページ内容のPDL記述をデコンポーザ処理してページイメージを生成する(S60)。ここで付与するページIDも、例えばGUIDを用いることにより一意性を保証することができる。このようにデコンポーザ25でページIDを付与する仕組みを設けたことにより、クライアント装置10でページIDが付与されなかったページについても、クライアント装置10で付与されたページと同じ扱いが可能となる。当該ページのデコンポーザ処理が完了すると、デコンポーザ25は、作成したページイメージとページIDとをイメージ形成装置24に渡し、1ページ分のページイメージの作成処理の完了を通知する(S62)。

【0052】<ページID記述があった場合>ページID記述を検出した場合は、デコンポーザ25は、そのページIDをイメージ形成装置24に渡し、そのページIDに対応したページイメージファイルがページキャッシュ26に保存されているかどうかの問合せを行う。そして、デコンポーザ25は、PDLファイルの処理を中断し、イメージ形成装置24からこの問合せに対する回答が返ってくるのを待つ。

【0053】イメージ形成装置24は、この問合せに応じ、そのページIDを持つページイメージの有無を調べるための検索依頼をページキャッシュ26に対して行う(S36)。

【0054】ページキャッシュ26は、検索依頼されたページIDを保存しているかどうか、検索処理を行う(S70)。各ページイメージを、ページIDをファイル名として保管していれば、この検索処理はファイル名の検索でよい。そして、ページキャッシュ26は、検索の結果、すなわち当該ページIDに対応するページイメージが有ったか無かったかのいずれかを示す信号をイメージ形成装置に返す(S72)。

【0055】ここで、ページキャッシュ26からそのページIDを持つページイメージのファイルが見つからなければ、このページは、初めて送られてきたジョブの中の1ページであるか、あるいは以前送られてきたジョブの中の修正されたページか、あるいはページキャッシュ26からすでに削除されてしまったページか、のいずれかである。いずれにせよ、このページのイメージはページキャッシュ26に存在しないので、PDLファイルからデコンポーザ処理してページイメージを作成する必要がある。そこで、イメージ形成装置24は、デコンポーザ25に対して、問合せに対する回答として、該当ペー

ジイメージが存在しなかった旨を通知する (S38)。

【0056】この通知を受けたデコンポーザ25は、中断していた処理を再開し、ページIDの作成処理 (S58) をスキップして当該ページについてのデコンポーズ処理を行い、ページイメージを作成する (S60)。そして、当該ページのデコンポーズ処理が完了すると、デコンポーザ25は、作成したページイメージとページIDとをイメージ形成装置24に渡し、1ページ分のページイメージの作成処理の完了を通知する (S62)。なお、この場合のページIDは、PDLファイルから検出されたページIDである。

【0057】一方、S70の検索でページキャッシュ26中に当該ページIDを持つページイメージが見つければ、以前にデコンポーズ処理したことのあるページであるということであり、イメージ形成装置24はデコンポーザ25に対し、該当ページイメージが存在した旨を通知する (S38)。

【0058】この通知を受けたデコンポーザ25は、ページIDの作成 (S58) 及びデコンポーズ処理 (S60) をスキップし、イメージ形成装置24に対し、1ページ分のページイメージの作成処理の完了を通知する

(S62)。すなわち、例えば、デコンポーザ25は、ページの区切りを表すコメント文などから次のページの先頭を検出し、そこまでのPDL記述の処理を省略する。なお、このケースでは、デコンポーザ25はページイメージの作成を行わないので、ページIDのみを完了通知と共にイメージ形成装置24に返す。

【0059】<ページイメージ作成完了通知後>デコンポーザ25は、S62でページイメージの作成完了をイメージ形成装置24に通知した後、PDLファイルに次のページに関する記述があるかどうか調べ (S64)、あった場合はS52に戻って以上の処理を繰り返す。

【0060】一方、デコンポーザ25からS62の完了通知を受けたイメージ形成装置24は、その時受け取ったページIDをページIDリストに登録する (S42)。

また、イメージ形成装置24は、上記完了通知と共にページイメージをデコンポーザ25から受け取った場合は、そのページイメージをページIDと共にページキャッシュ26に渡し、格納を依頼する (S44)。これを受けたページキャッシュ26は、そのページイメージとページIDとを対応づけて保存する (S74)。この場合、イメージ形成装置24は、1ページのページイメージを1つのファイルとし、ファイル名を付与して保存する。ファイル名としては、当該ページのページIDを用いることが、再印刷時などにおける検索処理の観点などからみて好適である。なお、ページイメージのファイル名をページID以外のものとしてももちろんよいが、この場合、各ファイルには、後の検索に備えてページIDの情報を含めておく。例えば、ページキャッシュに保存されるページイメージファイルのファイル名とし

てページIDをそのまま利用した場合、構成ページファイルは図4に示すように記述される。図4では、1行目が第1ページ、2行目が第2ページ、といった具合に、各行が各ページのページイメージのファイル名 (すなわちページID) を示す。

【0061】なお、S70のページキャッシュ26の検索結果が、「該当ページイメージ有り」である場合、S44及びS74の処理は不要である。

【0062】<全ページの処理完了後>以上が、1ページ分のページイメージの作成処理である。S52以降、PDLファイルを順に解釈して以上の処理を繰り返すことにより、ジョブの各ページのページイメージが獲得できることになる。そして、S64において、デコンポーザ25がPDLファイルにおいて次のページの記述がないことを検出すると、全ページについてのページイメージの作成が終了する。このとき、デコンポーザ25は、イメージ形成装置24に対し全ページイメージの作成完了を通知する (S66)。

【0063】これを受けたイメージ形成装置24は、イメージ形成を行った各ページのページIDを登録したページIDリストに基づき構成ページファイルを作成し (S46)、この構成ページファイルを含む処理完了通知をジョブ管理装置22に送信する (S48)。これで、ジョブ管理装置22が依頼したジョブについてのイメージ形成処理が終了する。

【0064】(8) IOT制御装置27

以上のようにして、あるジョブの全ページのイメージ作成が完了すると、ジョブ管理装置22は、イメージ形成装置24から受け取った構成ページファイルとジョブチケットの情報とをIOT制御装置27に渡し、印刷を指示する。

【0065】これを受けたIOT制御装置27は、構成ページファイルに示されたアクセス情報 (ファイル名、ページIDなど) に基づき、ジョブの各ページのページイメージをページキャッシュ26から読み出して順次IOT30に送り、ジョブチケットの情報、例えば部数やフィニッシングなどの指定に基づきIOT30を制御して印刷を行わせる。

【0066】[まとめ] 以上、本発明の好適な実施の形態を説明した。以上の説明から分かるように、本実施形態では、クライアント装置10側でジョブに含まれる個々のページごとに一意的なページIDを付与し、プリントサーバ20側で各ページのページイメージをこのページIDと対応づけて保存することにより、印刷可能なイメージデータをページ単位で再利用することを可能とした。

【0067】本実施形態では、クライアント装置10では、ページIDを用いて文書の編集履歴をページ単位で管理し、各ページのページIDをジョブの中に埋め込んでプリントサーバ20に渡す。プリントサーバ20で

は、各ページごとに、以前に作成したページイメージがページキャッシュ26に保存されているかどうかをページIDを用いて調べ、保存されているページについてはそれを再利用する。このような連携により、再印刷などの場合に修正のなかったページについてはPDLからのページイメージ作成処理を省略することができ、再印刷処理の処理速度を向上させることができる。修正の全くない同じ内容の文書の再印刷の場合、基本的にPDLの解釈処理を全く行う必要がないので非常に高速に処理できる。このように、本実施形態によれば、プリントサーバ20での処理の量が低減され、サーバの全体的なスループットの向上を見込める。

【0068】また、本実施形態は、再印刷のみならず、複数ページのうち一部のページのみを次々に差し替えて多部数の文書を作成するという可変ページ印刷の用途にも威力を発揮する。例えば保険の契約書は、大部分のページは特定の顧客に依存しない共通の内容であり、ほんの数ページのみが契約者名や締結日などのような顧客依存のデータが入る。このような可変ページ文書を多部数印刷する場合に本実施形態を利用すれば、大部分のページについてはページキャッシュ26に保存されたページイメージを再利用できるので、全体の印刷処理時間を大幅に短縮できる。

【0069】また、本実施形態ではページIDをコメント文として組み込むので、プリントサーバ20は、ページIDを付与できないクライアントアプリケーションやプリンタドライバからもそのまま利用できる。ただし、この場合、クライアント装置側から再印刷を指示した場合には、ページIDを用いないためページキャッシュ26を活用できず、処理速度の向上は望めない。もっとも、この場合でも、プリントサーバ20側のジョブ保管装置23にアクセスしてジョブ単位で再印刷を指示することができ、そのジョブの構成ページファイルを利用して高速な印刷を実現することができる。

【0070】また、本実施形態では、従来のプリンタドライバが作成する全ページの内容を記述したPDLファイルに、各ページIDをコメント文として組み込むので、ページIDを認識できない従来のプリントサーバでもこのPDLファイルを解釈し、問題なく処理することができる。

【0071】〔第1変形例〕上記の実施形態では、クライアント装置10は文書のすべてのページを表すPDL記述を作成し、その中に各ページのページIDを埋め込んでいた。これは、上記実施形態では、従来のクライアント装置及び従来のプリントサーバとの互換性を考慮したため、プリントサーバが管理しているページキャッシュの情報をクライアント装置側で知ることができないことを前提としたためであった。このため、文書中のいくつかのページがプリントサーバ側のページキャッシュにページイメージファイルとしてすでに存在する場合で

も、クライアント装置では、すべてのページのPDLファイルを作成しなければならない、その結果PDLファイルにはプリントサーバ側で読み飛ばされることになる記述も含まれてしまっていた。

【0072】これに対し、この変形例では、クライアント装置10とプリントサーバ20との対話セッションにより、クライアント装置10側でプリントサーバ20のページキャッシュ26の情報を取得し、これをもとにPDL記述の量を削減する。

【0073】すなわち、クライアント装置10のプリンタドライバ14は、各ページのPDL記述を行うに先立ち、アプリケーション12から受け取ったそのページのページIDを持つページイメージが、すでにプリントサーバ20のページキャッシュ26中に存在するかどうかをプリントサーバ26に問い合わせる。これを受けたプリントサーバ20では、ジョブ管理装置22がページキャッシュ26を検索し、そのページIDを持つページイメージのファイルが保存されているかを調べ、その結果、すなわちページイメージの有無の情報をプリンタドライバ14に返す。

【0074】その回答が、該当ページイメージがページキャッシュ26中に存在する、というものであった場合は、プリンタドライバ14は、そのページのページ内容についてのPDL記述を行う代わりにページキャッシュ中のページイメージを使用する旨の指示を記述する。この記述は、上記実施形態と同様、コメント文の形式を用いて行うことができる。この記述における、ページキャッシュ中のページイメージの特定は、ページIDを用いればよい。一方、プリントサーバ20からの回答が、該当ページイメージがページキャッシュ中に存在しないというものであった場合は、プリンタドライバ14は、そのページの内容についてのPDL記述を従来どおりに行い、当該ページのページIDをコメント文、その他の形で付加する。

【0075】なお、プリンタドライバ14は、ジョブの送り先がページキャッシュ26を持たないプリントサーバだと分かった場合は、従来通りのPDL記述を行う。

【0076】プリントサーバ20は、クライアント装置10から受け取ったPDLファイルを先頭から順に解釈していき、ページキャッシュ26中の特定のページイメージを使用する旨のコメント記述を検出した場合は、その記述内で指定されているページIDをページIDリストに登録するようにすればよい。そのページIDを持つページイメージがページキャッシュ26に保存されていることは、PDLファイル作成時の対話セッションにより既に分かっているので、プリントサーバ20ではページキャッシュ26の検索を改めて行う必要はない。その他の処理については、上記実施形態と同様でよい。

【0077】このように、この変形例では、プリンタドライバ14で作成されるPDLファイルには、プリント

サーバ20のページキャッシュ26内にあるページについては、そのページの内容を表すPDL記述が省略される。このため、次のような様々な効果が得られる。

【0078】まずクライアント装置10側では、ページを記述するという負荷の高い処理が減り、作成するPDLファイルが小さくなることから、負荷が低減され、処理の高速化が見込める。例えば、以前印刷したときと同じ文書を、変更せずにまったく同じ状態で再度印刷しようとした場合、すべてのページのイメージは既にページキャッシュ26中に存在するので、作成されるPDLファイルのサイズは最小になり、最も効率的である。したがって、クライアント装置10のアプリケーション12及びプリンタドライバ14は、印刷処理から即座に開放される。

【0079】またネットワーク40について言えば、転送するPDLファイルのサイズが小さくなることから、負荷の低減、ファイル転送の高速化が見込める。

【0080】またプリントサーバ20側では、処理するPDLファイルが小さくて済むので、ジョブ保管装置23で必要となるディスク容量が少なくて済み、プリントサーバ20全体の処理負荷も低減され、結果的に高速化が見込める。

【0081】なお、対話セッションを持つことができない上記実施形態のクライアント装置10でも、上記実施形態と同じ処理を行うことで、この変形例のプリントサーバ20と共に問題なく印刷処理を行うことができる。

【0082】このように、本変形例によれば、システム全体で処理する情報量が減少し、システム各部での記憶容量や通信時間などの必要なリソースが少なくて済み、また処理負荷が軽くなり処理が高速化される。

【0083】〔第2変形例〕上記実施形態では、プリントサーバ20では、キャッシュするイメージデータの単位を文書のページとしてきた（ページキャッシュ26）。しかし、キャッシュ単位はなにもページに限定する必要はない。ここでは、イメージのキャッシュの単位の他の例を説明する。

【0084】まず、文書をキャッシュの単位とすることが可能である。例えば、Adobe Systems社のPDF (Portable Document Format) という文書形式では、ファイル内に文書自体の一意的なIDと、文書のバージョンごとの一意的なIDを入れることを必須としているので、各文書を各バージョンごとに一意に識別できる。これらのIDを利用し、プリントサーバ20で文書全体をキャッシュしてしまうという方法が考えられる。プリントサーバ20のイメージデータのキャッシュ装置では、文書を構成する全ページをまとめて管理するのである。PDFファイルのような内容の変更を前提としない文書では、文書ごとにキャッシュする方法は再印刷の高速化に非常に有効である。

【0085】また、文書中に含まれる個々の描画オブジ

ェクトをキャッシュの単位とすることも好適である。ここでいう描画オブジェクトとは、ページよりも小さな、文書の物理的な構成要素であり、例えば文字枠（テキスト）や図形枠（グラフィックス、連続階調イメージなど）などである。

【0086】この変形例では、クライアント装置10は、このような各描画オブジェクトにそれぞれ固有のIDを付与してPDLファイルに組み込み、プリントサーバ20側では、作成した各描画オブジェクトのイメージデータをそのIDと対応づけてキャッシュする。キャッシュするイメージデータには、当該描画オブジェクトのページ内での位置の情報を対応づけて保管する。クライアント装置10は、描画オブジェクトに対して内容変更、移動、拡大・縮小などの変更が加えられると、変更後のオブジェクトには新たなIDを付与する。

【0087】クライアント装置10から一部を修正した文書の再印刷のジョブが来た場合、プリントサーバ20では、このジョブの各描画オブジェクトのIDをPDLファイルから検出し、そのIDを持つイメージデータがキャッシュ装置に保存されているかどうかを調べ、保存されている場合には、そのイメージデータをキャッシュ装置から読み出して再利用する。そのIDのイメージデータがキャッシュされていない場合は、その描画オブジェクトのPDL記述について通常のデコンボーズ処理を行う。この場合、デコンボーズ25が、キャッシュ装置から読み出したイメージ及びデコンボーズ処理により生成したイメージを合成し、1ページのイメージを生成する。

【0088】このように、描画オブジェクト単位でイメージデータをキャッシュすれば、一度作成したイメージデータをより効率的に再利用することができ、再印刷処理を高速化することができる。また、ページよりも小さな単位でキャッシュすると、キャッシュの利用効率も上昇する。

【0089】なお、描画オブジェクト単位でキャッシュを行う構成では、ページ単位の場合より印刷時のキャッシュ検索の回数が多くなり、1回の検索に要する時間も長くなりがちである。この点は、各描画オブジェクトの性質を考慮することにより改良することができる。

【0090】例えば、PostScriptファイルの場合、テキストデータやベクトル描画データ（グラフィックス）などのオブジェクトに関しては、一般にデコンボーズ処理に大きな時間はかからない。これに対し、ラスタイメージのオブジェクトの場合、デコンボーズ処理に、解像度変換や色分解、ハーフトーン処理など、負荷が高く時間のかかる処理が多いので、処理に多大の時間を要する場合が多い。また、この一方で、文書作成の現場ではよく知られているように、テキストデータは頻繁に編集されて更新されることが多いが、ラスタイメージはテキストよりも編集される頻度は少ないことが多い。

【0091】このような描画オブジェクトの性質を考慮すると、デコンポーズ処理に時間がかかり、変更されることの少ないラスタイメージオブジェクトのキャッシュが有効であることが分かるであろう。テキストやグラフィックスについては、デコンポーズ処理に要する時間が短いので、キャッシュの必要性は低い。したがって、ラスタイメージの描画オブジェクトのみをキャッシュの対象とすることにより、プリントサーバ20におけるキャッシュ検索負荷の増大を避けつつ、デコンポーズ処理の負荷を軽減できる。

【0092】上述の実施形態では、ページ単位でイメージデータのキャッシュを行っていたため、ページ中にラスタイメージとテキストが混在する場合、テキストが変更されただけでもページ全体をデコンポーズ処理しなければならなかったが、この改良方式では、ラスト部分はキャッシュ内のイメージデータを利用することができるので、高速に処理することができる。

【0093】【第3変形例】ページキャッシュ26は一つのプリントサーバ20で利用するよりも、複数のプリントサーバ20の間で有機的に結合して利用した方が、より効率がよい。有機的結合は、ネットワーク40を介して互いに通信可能なプリントサーバ20どうしで、ページキャッシュ26の情報を交換することで行う。

【0094】この変形例では、イメージ形成装置24は、目的のページIDを持つページイメージがページキャッシュ26になかった場合は、他のプリントサーバ20に対し、そのページIDを持つページイメージがそのサーバ20のページキャッシュ26に保管されているかどうか問い合わせる。ネットワーク40上のいずれかのプリントサーバ20から当該ページイメージを保管している旨の回答が有れば、イメージ形成装置24は、そのプリントサーバ20の名前を記憶し、デコンボザ25にページイメージが存在していたことを伝える。デコンボザ25は、それに応じ、そのページのデコンポーズ処理をスキップする。

【0095】そして、イメージ形成装置24が作成する構成ページファイルには、上記実施形態に例示したページIDだけではなく、そのページイメージが保管されたプリントサーバの名前の情報も記録する。IOT制御装置27は、構成ページファイルに示されたページIDとプリントサーバ名とを用いてページイメージのファイルを取得し、IOT30に供給する。

【0096】この構成によれば、印刷システムに含まれる複数のプリントサーバ20のいずれかにページイメー

* ジが保管されていれば、それを再利用することができる。したがって、各プリントサーバ20ごとに同じページイメージを重複して保存する必要がなくなる。また、各プリントサーバ20が、すべてのプリントサーバ20のページキャッシュ26を等しく利用できるため、各プリントサーバ20のキャッシュ容量が、見かけ上、システムに属する全プリントサーバ20のキャッシュ総量に等しくなる。このように、この構成によれば、各プリントサーバ20においては、速いイメージ形成とキャッシュ容量の増加という効果が得られる。

【0097】以上、本発明の好適な実施の形態並びにその変形例を説明した。以上では、ページIDをコメント文としてPDLファイル中に埋め込んだが、本発明はこのような方式に限定されるものではない。例えば、各ページとページIDとの対応関係を示すテーブルを作成し、PDLファイルに対応づけてプリントサーバ20に送るような方式でも同様の効果を得ることができる。また、以上では、ジョブの情報（PDLファイル、ジョブチケット、構成ページファイルなど）を保管するジョブ保管装置23をプリントサーバ20の内部に設けたが、ジョブ保管装置23をプリントサーバ20とは別の装置としてネットワーク40に接続するような構成も本発明の範囲内である。この場合、ネットワーク40に接続した1つのジョブ保管装置23を複数のプリントサーバで共用することも可能である。

【図面の簡単な説明】

【図1】 本発明に係る印刷システムの概略構成を示す機能ブロック図である。

【図2】 ページIDが組み込まれたPDL記述の一例を示す図である。

【図3】 プリントサーバの処理手順を、ジョブ管理装置の処理を中心に示したフローチャートである。

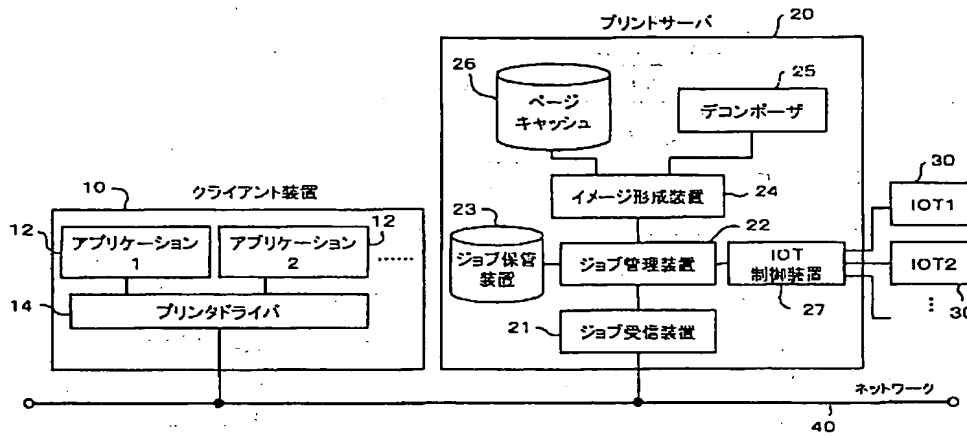
【図4】 構成ページファイルの内容の一例を示す図である。

【図5】 プリントサーバにおけるイメージ形成処理の手順を示すフローチャートである。

【符号の説明】

10 クライアント装置、12 アプリケーション、14 プリンタドライバ、20 プリントサーバ、21 ジョブ受信装置、22 ジョブ管理装置、23 ジョブ保管装置、24 イメージ形成装置、25 デコンボザ、26 ページキャッシュ、27 IOT制御装置、30 IOT。

【図1】



【図2】

```

...
restore
gsave showpage grestore
%%Page: 7 7
%%PageID: {5a61f7a0-cde1-11cf-9113-00aa00425c62}
save
...

```

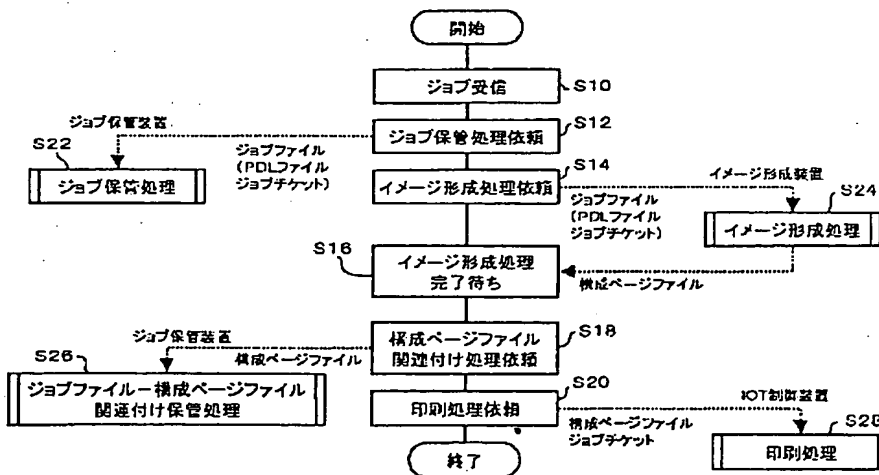
【図4】

```

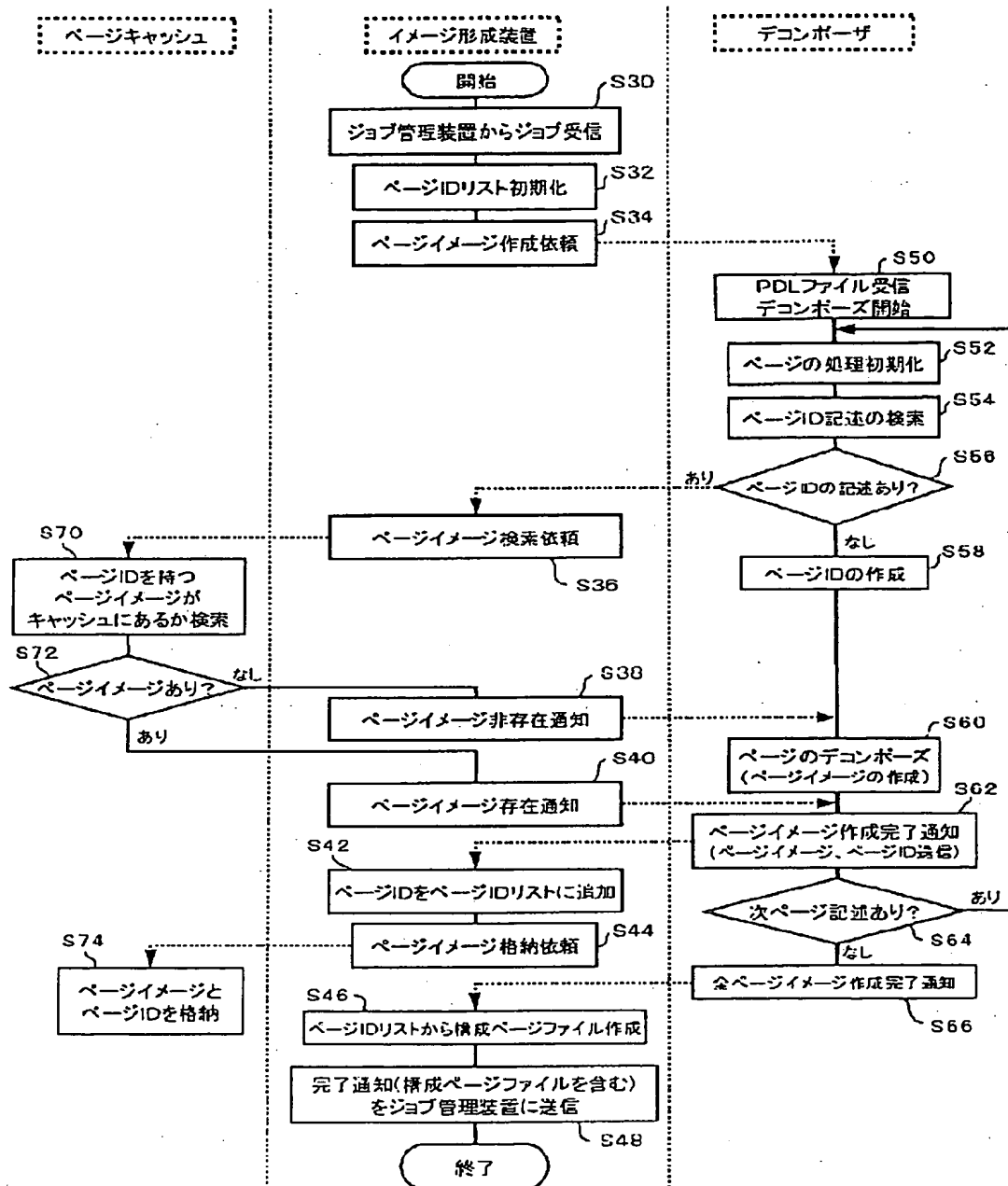
{5a61f7a0-cde1-11cf-9113-00aa00425c62}
{5a61f7a1-cde1-11cf-9113-00aa00425c62}
{5a61f7a2-cde1-11cf-9113-00aa00425c62}
{5a61f8f3-cde1-11cf-9113-00aa00425c62}
{5a61f7a4-cde1-11cf-9113-00aa00425c62}
{5a61f7a5-cde1-11cf-9113-00aa00425c62}
...

```

【図3】



【図5】



*** NOTICES ***

Japan Patent Office is not responsible for any damages caused by the use of this translation.

1. This document has been translated by computer. So the translation may not reflect the original precisely.
 2. **** shows the word which can not be translated.
 3. In the drawings, any words are not translated.
-

CLAIMS

[Claim(s)]

[Claim 1] The client equipment which creates the print job containing the printing directions data of at least one document component, A print job is interpreted, the data of each document component which can be printed are generated, and the print server which supplies to an airline printer and it is made to print is included. Said client equipment A means to give an unique identifier within said printing system at least to the document component changed [which changed and new-generated], It has the means which includes the correspondence-related information of the identifier and document component which were given in a print job. Said print server As opposed to the data which were generated from the printing directions data of each document component and which can be printed. The printing system characterized by making reusable the data which have a storage means to match and memorize the identifier given to the document component concerned, and were memorized by said print server using said identifier, and which can be printed.

[Claim 2] The means which includes the correspondence-related information of the identifier and document component in said client equipment in a print job is a printing system according to claim 1 characterized by incorporating in a print job in the form of the comment sentence to the printing directions data which described the document component which corresponds said identifier.

[Claim 3] A means to detect the identifier which said print server analyzed the comment sentence of a print job, and was given to each document component, A means to judge whether the data corresponding to the detected identifier which can be printed are memorized by said storage means, Instead of generating the data which can be printed from the printing directions data of the document component concerned, when judged with memorizing The printing system according to claim 2 characterized by having a means to use as data which read the data corresponding to said detected identifier which can be printed from said storage means, and were generated from the printing directions data, and which can be printed.

[Claim 4] A means by which the data of each document component with which said client equipment constitutes a print job to said print server which can be printed ask whether said storage means memorizes using the identifier of each document component, About the document component which received the reply that the data which can be printed are memorized by said storage means as a result of this inquiry The means which generates description of the purport which reads and uses the data corresponding to the identifier of

this document component memorized by said storage means which can be printed instead of creating the printing directions data which described the document component concerned, and is built into a print job, The printing system according to claim 1 characterized by ****(ing).

[Claim 5] Said printing system has said two or more print servers. Said each print server When there are no data corresponding to the identifier of the document component contained in the print job received from client equipment which can be printed in the storage means of self It asks whether have memorized the data corresponding to the identifier concerned which can be printed to other print servers. It is a printing system given in either of claim 1 to claims 4 characterized by receiving supply of the data concerned which can be printed from the print server when there is a print server which has memorized the data corresponding to the identifier concerned which can be printed.

[Claim 6] Said document component is a printing system given in either of claim 1 to claims 5 characterized by being a page.

[Claim 7] From the document data which the application which gives the identifier of a proper for each [which constitutes a document] document component of every, and manages the updating hysteresis created A means to be the printer driver which generates the print job of the predetermined Page Description Language which described the printing image of the document, and to extract the identifier of each document component of the document concerned from document data, A means to generate description of the Page Description Language showing said each document component, The printer driver which has a means to add to description of the Page Description Language showing said each document component in the form of the comment sentence in which the information on the identifier of the document component concerned was specified at the Page Description Language concerned, and to generate a print job.

[Claim 8] It has an interpretation means to interpret the print job described by the Page Description Language, and to generate the data which can be printed. It is the print server which makes an airline printer supply and print the generated data which can be printed. A storage means to be a storage means to memorize the data which were generated with said interpretation means and which can be printed, and to match and memorize the data which can be printed and the identifier of a document component of the document component for every document component of a document, A means to analyze a print job and to detect the identifier of each document component of a document which the print job concerned expresses, When judged with a means to judge whether the data corresponding to the detected identifier which can be printed are memorized by said storage means not being memorized When said interpretation means is made to interpret the printing directions data of the document component corresponding to said detected identifier, the data of the document component concerned which can be printed are generated and it is judged with memorizing A means to read the data corresponding to said detected identifier which can be printed from said storage means, and to output them, The print server which has a means to make said storage means memorize the identifier and the data which can

be printed of the document component, about the document component which generated the data which can be printed with said interpretation means.

DETAILED DESCRIPTION

[Detailed Description of the Invention]

[0001]

[Field of the Invention] This invention interprets the printing directions data generated with client equipment by the print server, generates the data which can be printed, and relates to the technique for mitigating the processing load of re-printing in the printing system which makes an airline printer supply and print this.

[0002]

[Description of the Prior Art] In order to perform re-printing efficiently, the documentation-management equipment which saves and manages a document may be formed between a print server and client equipment or into a print server. In such a system, when the need for re-printing of a certain document arises, the time and effort which generates the printing directions data of the again same document with client equipment, and is transmitted to a print server by taking out and printing the data of an identical text document saved at the time of former printing from documentation-management equipment is saved, and the effectiveness of re-printing is raised.

[0003] Now, when a series of processings from document edit to printing are considered, it is known well that one document will be changed into suitable various file formats, and will be treated in each processing phase. There are strengths and weaknesses in each format, respectively, and it becomes the important point whether what we do with the file format of the document kept with documentation-management equipment, when considering the effectiveness of a system.

[0004] Although that of the file format (or close to it) of document edit application is suitable for edit of a document, a problem has the case where it is a format of an application proper, in respect of versatility by most. Moreover, the specification of a file format itself may not be exhibited and, in such a case, construction of document storage equipment becomes difficult primarily. Moreover, when time amount is required by completion from directions of re-printing when the file format of document edit application is used, or a different airline printer from the time of the first printing is specified as an output destination change, the same printing result may not be obtained from the inequality of a font etc., and this format is hard to be referred to as suitable for the format of the storage data for re-printing.

[0005] The file format (for example, raster image) finally used inside a printer on the other hand will not usually be able to become original with the printer (or printer vendor), and cannot be used general-purpose by other printers etc. When the system configuration which connects various printers with which various vendors provide a print server is

considered, such a format is not suitable for the document storage format for re-printing. [0006] Since it is such, with documentation management equipment aiming at re-printing, PDL (Page Description Language) is used as a format of the document to keep in many cases. PDL is language which describes a page and, generally is used as a format at the time of the document data transfer from client equipment to a print server. PDL has some which are supported by the print server of many vendors, such as industry standard PostScript (trademark of U.S. Adobe Systems), aiming at the page description independent of the model of each devices, such as a printer. remarkable in the identity of a re-printing result when PDL is used as a format for storage of a document and it can use general-purpose by various printers -- an extent guarantee can be offered and the print speed at the time of re-printing is also improved compared with a format of document edit application.

[0007] However, PDL takes a certain amount of time amount to change into the raster format which can be processed by the printer. The expansion to a raster image from PDL takes quite long time amount to a document including especially a color image as everyone knows. For this reason, the improvement of a re-print speed is demanded in the system which keeps a document in a PDL format.

[0008] The system by which it not only keeps a document in the form of a PDL file, but the file of a format (referred to as PRF (print ready format)) which specialized in printing by the specific printer compares and keeps it as a system which meets this request exists. PRF is the format (generally a kind of a raster) to which resolution, an image main scanning direction, etc. were set so that this printer might work most efficiently, and as a result of carrying out expansion processing of the PDL file, it is usually created. In such a system, when the last time same printer as the time of printing performs re-printing of a document, re-printing processing can be performed at a high speed by using the PRF file of the document concerned which matched with PDL and was kept.

[0009]

[Problem(s) to be Solved by the Invention] however, ** while the system is performing the documentation management as a unit of management of a PDL file such conventionally and correction of some document is made with an application level -- when the event which may be generated frequently happens, the inefficient situation arises. For example, when a miswritten word was discovered and corrected after printing on application, the PRF file of the document concerned before the correction which created even correction of only one character before since the PDL file of the whole document will be re-created and the PDL file itself was changed in that case could not be used, but late expansion processing of processing needed to be again carried out to the re-created PDL file.

[0010] That is, although the documentation management of the above-mentioned conventional PDL base was suitable for managing the completed document without the correction which carried out printing at hand, in the environment where the document with which correction is made frequently is dealt with, it could not use the PRF file made with much trouble before, and was not able to obtain merits, such as improvement in the

speed of re-printing.

[0011] In the system which is made in order to solve the above problem, keeps the data of the document printed once, and is used for re-printing, this invention uses effectively the data of the document printed before also in the environment where document correction is made frequently, and aims at improving the processing speed of re-printing after correction.

[0012]

[Means for Solving the Problem] In order to attain the above-mentioned purpose, the printing system concerning this invention The client equipment which creates the print job containing the printing directions data of at least one document component, A print job is interpreted, the data of each document component which can be printed are generated, and the print server which supplies to an airline printer and it is made to print is included. Said client equipment A means to give an unique identifier within said printing system at least to the document component changed [which changed and new-generated], It has the means which includes the correspondence-related information of the identifier and document component which were given in a print job. Said print server It has a storage means to match and memorize the identifier given to the document component concerned, to the data which were generated from the printing directions data of each document component and which can be printed, and is characterized by making reusable the data which were memorized by said print server using said identifier and which can be printed.

[0013] With this configuration, it became possible to reuse the data which can be printed not in the unit of a PDL file but in the unit of a document component by giving an unique identifier for every document component of each which is contained in a print job by the client equipment side, matching with this identifier the data of each document component which can be printed, and saving them by the print server side. A document component may be a page and may be drawing objects, such as a discrete character train, and a graphic form, an image. With this configuration, also in one document, an identifier different correction before is given to the corrected document component, and the same identifier as origin is given to the document component which was not corrected. Therefore, when it corrects to the document printed once, about the document component which is not corrected, the data which were used for printing and which can be printed can be reused and printed in front, and the processing which develops printing directions data to the data which can be printed can be excluded.

[0014] In the suitable mode of this invention, client equipment is incorporated in a print job in the form of the comment sentence to the printing directions data which described the document component which corresponds an identifier. A comment sentence is the format of not affecting the generation means (the so-called interpreter) of the data which a print server has and which can be printed. The data with which the data which can be printed already correspond to it about a document component [finishing / storage] at a storage means based on the detected identifier if a means to analyze the comment sentence in a print job other than an interpreter, and to detect an identifier is formed in a print server

and which can be printed can be searched and used. Moreover, in this mode, since the printing directions data which described each document component are contained in the print job, even if it processes by the print server which does not correspond even if it processes by the print server corresponding to this invention, the same printing result is obtained fundamentally. Therefore, this mode can be used without conflict, even when all the print servers on a network do not have the structure using the identifier of this invention, and it can be easily introduced also into the network environment to which many print servers were connected.

[0015] In another suitable mode of this invention, moreover, client equipment Whether the data of each document component which constitutes a print job which can be printed are memorized by the storage means to the print server About the document component which received the reply that it asks using the identifier of each document component and the data which can be printed are memorized by the storage means as a result of this inquiry Instead of creating the printing directions data which described the document component concerned, the printing directions data which direct the purport which reads and uses the data corresponding to the identifier of this document component memorized by the storage means which can be printed are generated, and it includes in a print job.

[0016] According to this mode, the amount of data of a print job can be lessened and the time amount which that creation and transmission take can be reduced.

[0017] In the printing system which has said two or more print servers in another suitable mode of this invention moreover, each print server When there are no data corresponding to the identifier of the document component contained in the print job received from client equipment which can be printed in the storage means of self It asks whether have memorized the data corresponding to the identifier concerned which can be printed to other print servers, and when there is a print server which has memorized the data corresponding to the identifier concerned which can be printed, supply of the data concerned which can be printed is received from the print server.

[0018] Also in the system which has two or more print servers, by utilizing the image which was generated before and which can be printed as effectively as possible, the time amount which the interpretation of a print job takes can be reduced, and, according to this mode, the processing time of re-printing can be shortened.

[0019] moreover, this invention from the document data which the application which gives the identifier of a proper for each [which constitutes a document] document component of every, and manages the updating hysteresis created A means to be the printer driver which generates the print job of the predetermined Page Description Language which described the printing image of the document, and to extract the identifier of each document component of the document concerned from document data, A means to generate description of the Page Description Language showing said each document component, The printer driver which has a means to add to description of the Page Description Language showing said each document component in the form of the comment sentence in which the information on the identifier of the document component concerned was specified at the

Page Description Language concerned, and to generate a print job is offered.

[0020] Moreover, this invention has an interpretation means to interpret the print job described by the Page Description Language, and to generate the data which can be printed. It is the print server which makes an airline printer supply and print the generated data which can be printed. A storage means to be a storage means to memorize the data which were generated with said interpretation means and which can be printed, and to match and memorize the data which can be printed and the identifier of a document component of the document component for every document component of a document, A means to analyze a print job and to detect the identifier of each document component of a document which the print job concerned expresses, When judged with a means to judge whether the data corresponding to the detected identifier which can be printed are memorized by said storage means not being memorized When said interpretation means is made to interpret the printing directions data of the document component corresponding to said detected identifier, the data of the document component concerned which can be printed are generated and it is judged with memorizing A means to read the data corresponding to said detected identifier which can be printed from said storage means, and to output them, The print server which has a means to make said storage means memorize the identifier and the data which can be printed of the document component, about the document component which generated the data which can be printed with said interpretation means is offered.

[0021]

[Embodiment of the Invention] Hereafter, the gestalt (henceforth an operation gestalt) of operation of this invention is explained based on a drawing.

[0022] Drawing 1 is the functional block diagram showing the outline configuration of the printing system concerning this invention. A printing system consists of IOT(s) (Image Output Terminal)30 connected to the client equipment 10 and the print server 20 which were mutually connected through the network 40, and the print server 20 by the exclusive cable etc.

[0023] Client equipment 10 is a computer and carries various kinds of applications 12. If printing directions are performed from a user to the document drawn up with these applications 12, a printer driver 14 will change the document of the data format of application 12 into the data format of a Page Description Language (PDL), will generate a print job, and will transmit to a print server 20. A print server 20 is changed into data which can be printed, such as a raster image which interprets this print job and can be treated to IOT30. IOT30 is also called a print engine and forms the visible image of a document on a form based on the data which are supplied from a print server 20 and which can be printed. Two or more IOT(s)30 are connectable to one print server 20. Moreover, two or more client equipment 10 and print servers 20 are also connectable with a network 40.

[0024] Hereafter, the internal configuration of client equipment 10 and a print server 20 is explained in more detail.

[0025] [Client equipment]

(1) The application 12 of application 12 client equipment 10 gives an identification number (it is called Page ID) peculiar to each page of a document, and has the function to manage the revision history of each page. If the page is changed from the saved condition while Page ID is given to it, when newly generated by each page, the new page ID will be given to the page. The page ID of each page is contained in the document data which application 12 creates. In addition, Page ID is given so that it may become unique inside the printing system concerned at least.

[0026] GUID (Global Unique ID) which for example, Microsoft uses can be used for page ID grant. By entitling [ID / of the equipment itself] the ID, this is an algorithm which creates ID of the data, and can create 128-bit unique ID globally almost certainly while it gives ID of a proper for every equipment, respectively and gives unique ID within the equipment of opposite *Perilla frutescens* (L.) Britton var. *crispa* (Thunb.) Decne. to the data which each equipment generated. MS which is the word processor application of for example, Microsoft as application using GUID There is Word (trade name) and this can also be used as application 12 in this operation gestalt. In addition, as long as the algorithm of page ID grant is not restricted to this and can guarantee uniqueness within the printing system concerned, what kind of thing is sufficient as it.

[0027] (2) Printer driver 14 printer driver 14 cooperates in the business with application 12, and creates a print job (only henceforth a job). A job consists of a PDL file which described the printing image of each page of a document by PDL, and a job ticket file which described assignment of printing dispose attributes, such as number of copies and finishing processing (after treatment, such as a staple stop). In addition, a job ticket file is not essential to this invention, and it will be clearly understood from subsequent explanation that this invention can be applied also to the system which generates the print job which does not include a job ticket file.

[0028] While creating the PDL description as which a printer driver 14 expresses the printing image of each page of the document from the document data of application 12 in creation of the PDL file of a print job, the page ID of each [these] page is detected from document data, and it incorporates into description of PDL. The inclusion to the PDL file of Page ID can be performed by using the format of for example, a comment sentence. When PostScript (trademark of U.S. Adobe Systems) is used as PDL, it becomes description as shown in drawing 2 . the comment sentence to which the line which starts in "%%PageID:" expresses Page ID with this example -- it is -- { -- an inside value is Page ID. In addition, in PostScript, the line which starts with a notation "%" is recognized to be a comment, and an interpreter does not interpret this. "%%Page:77" The becoming description is a comment sentence which shows the beginning of the PDL description which described a certain 1 page (this example 77 pages) image, and the page ID of the next line is given to this page. Thus, in this example, description of Page ID is embedded into the head part of PDL description of the page concerned.

[0029] According to this method of incorporating during PDL description by making Page ID into a comment sentence, by the print server 20 side, in order to detect and analyze

description of Page ID, it is not necessary to add modification to the interpreter of PDL itself at all that what is necessary is just to establish the device (program) in which analyze a comment sentence and Page ID is extracted apart from the interpreter of PDL. Moreover, since a comment sentence is disregarded by the interpreter of PDL and it does not have effect of what on an interpreter, either, there is also an advantage of not losing compatibility to a print server without the device in which the comment sentence of Page ID is analyzed in this method.

[0030] By this method, a printer driver 14 cooperates in the business with application 12, creates the PDL description showing all the pages of the range directed to the user, and embeds the page ID of each page there as a comment sentence. Although the page by which had already been printed by the print server 20 may be that (it mentions later for details) which reuses the data which were then created, and which can be printed and it may be visible to futility apparently with this operation gestalt to create the PDL file containing all pages, this is for enabling it to print satisfactory also by the system which cannot recognize Page ID. That is, even when the print server 20 of this operation gestalt which can recognize Page ID on a network 40, and the print server which cannot recognize Page ID are intermingled with this method, a printer driver 14 can create the completely same PDL file, without being conscious of such mixture.

[0031] According to the job where such a page ID was embedded, a print server 20 can distinguish whether it is the thing which is that each page already printed once and which was received for the first time, or it will not be. For example, he can understand that only Pages X and Y were changed by already transmitting the job of the version A of a certain document 1 to a print server 20, and investigating the page ID of each page in a print server 20, when the version B of the document 1 with which only Pages X and Y were changed by the client equipment 10 side is created and it is sent to a print server 20 in the situation which has also completed printing. Although the page ID of other pages is the same as the time of an earlier version, only pages X and Y are because the earlier version has a different strange page ID.

[0032] Thus, the PDL file created by the printer driver 14 is transmitted to a print server 20 through a network 40 with the job ticket file created separately as a job.

[0033] [Print server]

(1) In job receiving set 21 print server 20, the job receiving set 21 carries out job reception from a network 40. The received job is sent to job management equipment 22, and is managed.

[0034] (2) Job management equipment 22 job management equipment 22 keeps the received job, and carries out generalization control of the whole flow of printing. The procedure of job management equipment 22 is explained with reference to drawing 3.

[0035] Job management equipment 22 will request storage of delivery and these files from job storage equipment 23 for the job file which received first, i.e., a PDL file, and a job ticket file, if a job is received from the job receiving set 21 (S10) (S12). Job storage equipment 23 keeps these files according to the request (S22.). It mentions later for details.

Moreover, job management equipment 22 manages the processing sequence of these receiving job. That is, job management equipment 22 registers and carries out scheduling of the received job to the queue of the waiting for printing. And if the sequence which prints a job comes, creation of delivery and a page image will be requested from image formation equipment 24 for the job (S14). Job management equipment 22 will be in the condition of the waiting for completion of image formation processing after this request (S16). Image formation equipment 24 controls deconstructivism POZA 25 and a page cache 26 according to this request, creates the file (it calls a configuration page file) which showed the access information (file name of each page image etc.) to each page image saved into these page cache 26, and passes it to job management equipment 22 while it saves the page image which created and created the page image of each page of that job into a page cache 26. Job management equipment 22 requests correlation processing with the file (a PDL file and job ticket file) of delivery and the original job from job storage equipment 23 for this configuration page file (S18). According to this request, job storage equipment 23 matches a configuration page file with a job file, and keeps it (S26.). It mentions later for details. And job management equipment 22 requests delivery and printing processing from the IOT control unit 27 for the configuration page file with the information on a job ticket (S20). The IOT control device 27 takes out the page image of each page from the page cache 26 with reference to a configuration page file, and makes it print by passing IOT30 according to this request (S28).

[0036] (3) Job storage equipment 23 job storage equipment 23 is a kind of database which keeps the information on each job for printing or re-printing. The job file (namely, a PDL file and a job ticket file) which received from client equipment 10, and the configuration page file which is the correlation information on each page image and job which are kept by the page cache 26 are contained in the information kept here. Since a series of information about a job is saved to job storage equipment 23, re-printing of the job can be directed to job management equipment 22 by accessing job storage equipment 23 from the user interface or the client equipment 10 of a print server 20, and specifying a job there. Since the page image of each page of the job concerned is already saved into the page cache 26 in re-printing, it is not necessary to interpret a PDL file anew and to perform image creation.

[0037] (4) Image formation equipment 24 image formation equipment 24 is equipment which changes the document of a job into the data [IOT30 / handling / data / which can be printed], and, and. Image formation equipment 24 receives a request of image formation processing of a job from job management equipment 22, and creates the page image which is data of each page which can be printed according to the PDL file and job ticket file which were received to coincidence at this time.

[0038] here -- being careful -- I hear that it is deconstructivism POZA 25 to perform [which it is called a PDL interpretation and drawing processing] processing for actual image creation, and there is. If it says in the viewpoint of image creation, image formation equipment 24 will request image creation to deconstructivism POZA 25, and will function as high order equipment which manages the whole image creation processing. The image

formation equipment 24 as such a management tool is formed for realizing the increase in efficiency using the page cache 26 of image formation processing.

[0039] With this operation gestalt, this processing is performed according to the page cache inquiry from deconstructivism POZA 25. Image formation equipment 24 investigates whether the page image of each page is finishing [storing] in the page cache 26 already using the page ID included in the PDL file. And when it becomes clear to make deconstructivism POZA 25 create the page image of the page, and to already be saved into the page cache 26, when it becomes clear that the page image of each page of a job is not saved into the page cache 26, image formation equipment 24 makes deconstructivism POZA 25 omit creation of the image of the page, and uses the page image in the page cache 26. By repeating such processing about each page, the page image of all the pages of a job can acquire from either deconstructivism POZA 25 or the page cache 26.

[0040] If the page image of all the pages of a job can be created, image formation equipment 24 will return the information on the page image of each page which constitutes a job as a response of an image creation request to job management equipment 22. With this operation gestalt, all the page images created by deconstructivism POZA 25 are saved into the page cache 26, and information required in order to access to the image of each [these] page in the page cache 26, i.e., a configuration page file, is passed to job management equipment 22.

[0041] The procedure of processing of this single string is explained in full detail by next (7) "the procedure of image formation processing."

[0042] (5) Deconstructivism POZA 25 deconstructivism POZA 25 interprets the PDL file of a job, creates the raster image of each page (this interpretation and image creation processing are doubled and it is called "deconstructivism pause processing"), and returns it to image formation equipment 24. In addition to the interpreter which interprets description of PDL, deconstructivism POZA 25 analyzes a comment sentence, detects Page ID, and has a device for asking whether the page image corresponding to the ID is already saved into the page cache 26. Deconstructivism POZA 25 investigates description of a PDL file sequentially from the head, and if it detects, an inquiry will be performed [whether the page-image file corresponding to the page ID which is Page ID is saved into the page cache 26 and] to image formation equipment 24. And when it turns out by this inquiry that the page image of that page ID is already saved in the page cache 26, the deconstructivism pause processing about that page is omitted. In addition, the detailed function of deconstructivism POZA 25 is explained in full detail by next (7).

[0043] (6) Keep as a file the page image of each page created from the PDL file by the deconstructivism pause processing by deconstructivism POZA 25, and the page cache 26 page cache 26 manages it. Here, the information on the page ID of the page concerned is related with the file of a page image. For example, it is suitable to make page ID itself into the file name of the page image concerned.

[0044] Since Page ID is unique, it is not necessary to manage especially the file of each page image for every job, and it can manage the image of all the pages of all jobs to

homogeneity on the same level. For example, one directory on a file system can be made into the page cache 26, and all page-image files can be managed.

[0045] When Page ID is used for the file name of a page image, the page-image file of a page with the page ID {5a61f8f3-cde1-11cf-9113-00aa 00425c62} is kept by the file name of {5a61f8f3-cde1-11cf-9113-00aa 00425c62}. Thus, what is necessary is just to search the file name in the directory which constitutes the page cache 26, in case the page-image file which has the specific page ID according to the inquiry from deconstructivism POZA 25 will be searched, if Page ID is used as a file name of the page-image file in a page cache.

[0046] In addition, the method which instead arranges and manages a page-image file for every job is also considered. For example, it is creating a subdirectory for every job all over the directory of the page cache 26, and saving the page-image file of the job under the subdirectory etc. This method of retrieval nature is better, and when re-printing directions in a job unit are received, this one can respond to a high speed.

[0047] The page cache 26 is realized using large capacity storage typically like a hard disk drive unit. Depending on the capacity of such a store, since the page cache 26 is the magnitude of finite, it cannot save a page-image file to infinity. For this reason, it is necessary to have the function to delete an unnecessary page-image file suitably. This function is realizable using well-known various technique in general cache memory. For example, when the total amount of a page image exceeds the upper limit defined beforehand, it is the method of deleting from the page image created in ancient times, or deleting from the page image which has not been referred to most for a long time. Moreover, sequential creation of the statistical data of the use frequency of a page image may be carried out, and the page image deleted based on this statistical data may be decided. Moreover, when the total amount of a page image exceeds a upper limit, structure which takes out warning to a server manager and a manager is made to delete is also possible.

[0048] (7) the procedure of image formation processing -- here, explain the flow of the image formation processing by the image formation equipment 24, deconstructivism POZA 25, and the page cache 26 which were explained until now with reference to drawing 5.

[0049] As for <initiation of processing> image formation processing, image formation equipment 24 begins from the time of receiving the processing request of a job from job management equipment 22. If image formation equipment 24 receives a job file (a PDL file and job ticket file) from job management equipment 22 and the processing request is received (S30), it will initialize a page ID list first (S32). This list will be used for creation of a configuration page file, and sequential registration of the page ID of each page which carried out image formation by subsequent processings will be carried out. And image formation equipment 24 takes out information required for image formation from a job ticket file, and requests creation of delivery and a page image from deconstructivism POZA 25 for this information with a PDL file (S34).

[0050] <Processing of deconstructivism POZA> deconstructivism POZA 25 will start deconstructivism pause processing, if those information is received (S50). Deconstructivism POZA 25 initializes various kinds of status information and drawing

memory which are first used for formation of a page image (S52), reads a PDL file sequentially from a head, and processes it. At this time, the existence of description of Page ID is inspected first (S54). In this operation gestalt, the page number of the page concerned and Page ID are incorporated in the form of a comment sentence before PDL description of the contents of a page of each page as mentioned above. The existence of description of this page ID is investigated in S54. After responding to the existence of page ID description, processing changes.

[0051] When description of Page ID <is not able to be detected [there is no page ID description and]> (i.e., when description of the contents of a page has started without description of Page ID in the PDL file), deconstructivism POZA 25 creates the page ID to the page concerned (S58), carries out deconstructivism pause processing of the PDL description of the contents of a page, and generates a page image (S60). Uniqueness can be guaranteed by using the pages ID and GUID given here. Thus, by having established the structure which gives Page ID by deconstructivism POZA 25, the treatment same also about the page to which Page ID was not given with client equipment 10 as the page given with client equipment 10 becomes possible. If deconstructivism pause processing of the page concerned is completed, deconstructivism POZA 25 will notify completion of creation processing of the page image and Page ID which were created of delivery and the page image for 1 page to image formation equipment 24 (S62).

[0052] When page ID description <is detected [there is page ID description and]>, deconstructivism POZA 25 performs the inquiry of whether delivery and the page image file corresponding to the page ID are saved to image formation equipment 24 in the page ID at the page cache 26. And deconstructivism POZA 25 interrupts processing of a PDL file, and waits for the reply to this inquiry to come on the contrary from image formation equipment 24.

[0053] Image formation equipment 24 performs the retrieval request for investigating the existence of a page image with that page ID according to this inquiry to the page cache 26 (S36).

[0054] The page cache 26 performs [whether the page ID by which the retrieval request was carried out is saved, and] retrieval processing (S70). If Page ID is kept for each page image as a file name, this retrieval processing is good at retrieval of a file name. And the signal which shows whether the page cache 26 was as a result of retrieval (i.e., the page image corresponding to the page ID concerned) or there was nothing is returned to image formation equipment (S72).

[0055] If the file of the page image which has that page ID from the page cache 26 is not found here, this page is 1 page in the job sent for the first time, or is in the page on which it was corrected in the job sent before or the page already deleted from the page cache 26, and ***** Anyway, since the image of this page does not exist in the page cache 26, it is necessary to carry out deconstructivism pause processing from a PDL file, and it needs to create a page image. Then, image formation equipment 24 notifies the purport that an applicable page image did not exist, as a reply to an inquiry to deconstructivism POZA 25

(S38).

[0056] Deconstructivism POZA 25 which received this notice resumes the interrupted processing, skips creation processing (S58) of Page ID, performs deconstructivism pause processing about the page concerned, and creates a page image (S60). And if deconstructivism pause processing of the page concerned is completed, deconstructivism POZA 25 will notify completion of creation processing of the page image and Page ID which were created of delivery and the page image for 1 page to image formation equipment 24 (S62). In addition, the page ID in this case is the page ID detected from the PDL file.

[0057] On the other hand, if the page image which has the page ID concerned in the page cache 26 by retrieval of S70 is found, I will hear that it is the page which had carried out deconstructivism pause processing before, it will be, and image formation equipment 24 will notify the purport that the applicable page image existed, to deconstructivism POZA 25 (S38).

[0058] Deconstructivism POZA 25 which received this notice skips creation (S58) of Page ID, and deconstructivism pause processing (S60), and notifies completion of creation processing of the page image for 1 page to image formation equipment 24 (S62). That is, for example, deconstructivism POZA 25 detects the head of the following page from the comment sentence showing a page break etc., and omits processing of the PDL description to there. In addition, in this case, since deconstructivism POZA 25 does not create a page image, Page ID is returned to image formation equipment 24 with the notice of completion.

[0059] After it notifies the completion of creation of a page image to image formation equipment 24 by S62, when it investigates whether there is any description about the following page to a PDL file (S64) and is in it, deconstructivism POZA 25 <after the notice of the completion of page image creation> returns to S52, and repeats the above processing.

[0060] On the other hand, the image formation equipment 24 which received the notice of completion of S62 from deconstructivism POZA 25 registers into a page ID list the page ID then received (S42). Moreover, image formation equipment 24 requests delivery and storing from the page cache 26 for the page image with Page ID, when a page image is received from deconstructivism POZA 25 with the above-mentioned notice of completion (S44). The page cache 26 which received this matches and saves the page image and Page ID (S74). In this case, image formation equipment 24 considers a 1-page page image as one file, and gives and saves a file name. As a file name, it is suitable to use the page ID of the page concerned, in view of the viewpoint of the retrieval processing in the time of re-printing etc. In addition, although it is easy to be natural also considering the file name of a page image as things other than Page ID, in preparation for next retrieval, the information on Page ID is included in each file in this case. For example, when Page ID is used as it was as a file name of the page-image file saved into a page cache, a configuration page file is described as shown in drawing 4. Drawing 4 shows the file name (namely, the page ID) of the page image each line of whose is each page to the condition which the 1st line called the 1st page and the 2nd line called the 2nd page.

[0061] In addition, when the retrieval result of the page cache 26 of S70 is "with an

applicable page image", processing of S44 and S74 is unnecessary.

[0062] The above <after the completion of processing of all pages> is creation processing of the page image for 1 page. The page image of each page of a job can be acquired by interpreting a PDL file in order and repeating the above processing after S52. And in S64, detection of that deconstructivism POZA 25 does not have description of the following page in a PDL file terminates creation of the page image about all pages. At this time, deconstructivism POZA 25 notifies the completion of creation of all page images to image formation equipment 24 (S66).

[0063] The image formation equipment 24 which received this creates a configuration page file based on the page ID list which registered the page ID of each page which performed image formation (S46), and the notice of the completion of processing containing this configuration page file is transmitted to job management equipment 22 (S48). Now, the image formation processing about a job which job management equipment 22 requested is completed.

[0064] (8) As it is 27 or more IOT control devices, if image creation of all the pages of a certain job is completed, job management equipment 22 directs delivery and printing for the configuration page file received from image formation equipment 24, and the information on a job ticket to the IOT control device 27.

[0065] The IOT control device 27 which received this reads the page image of each page of a job from the page cache 26, and makes it print to IOT30 one by one based on the access information (a file name, page ID, etc.) shown in the configuration page file by controlling IOT30 based on assignment of the information on delivery and a job ticket, for example, number of copies, finishing, etc.

[0066] The gestalt of suitable operation of this invention was explained more than the [conclusion]. With this operation gestalt, it made it possible to reuse the image data which can be printed per page by giving the unique page ID for every page of each which is contained in a job by the client equipment 10 side, matching the page image of each page with this page ID, and saving it by the print server 20 side, so that the above explanation might show.

[0067] With this operation gestalt, in client equipment 10, the edit hysteresis of a document is managed per page using Page ID, the page ID of each page is embedded into a job, and a print server 20 is passed. In a print server 20, it investigates using Page ID whether the page image created before is saved into the page cache 26 for every page, and it is reused about the page saved. By such cooperation, about the page which did not have correction in re-printing etc., the page image creation processing from PDL can be omitted, and the processing speed of re-printing processing can be raised. In re-printing of the document of the same contents of correction which is not, since it is not necessary to perform interpretation processing of PDL at all fundamentally, it can process at a high speed very much. Thus, according to this operation gestalt, the amount of processing by the print server 20 is reduced, and improvement in the overall throughput of a server can be expected.

[0068] Moreover, this operation gestalt demonstrates power not only for re-printing but for the application of adjustable page printing of substituting only some pages one after another among two or more pages, and drawing up the document of many number of copies. For example, the contracts of insurance are the common contents independent of the customer of specification [most pages], and the data of customer [pages / mere / several] dependence of a contractor name, a conclusion day, etc. enter. If this operation gestalt is used when carrying out multi-number-of-copies printing of such an adjustable page document, since the page image saved into the page cache 26 about most pages is reusable, the whole printing processing time can be shortened sharply.

[0069] Moreover, since Page ID is incorporated as a comment sentence with this operation gestalt, a print server 20 can be used as it is also from the client application and the printer driver which cannot give Page ID. However, when re-printing is directed from a client equipment side in this case, improvement in processing speed cannot be desired by that which can utilize the page cache 26 since Page ID is not used. But even in this case, the job storage equipment 23 by the side of a print server 20 can be accessed, re-printing can be directed per job, and high-speed printing can be realized using the configuration page file of that job.

[0070] Moreover, with this operation gestalt, since each page ID is included in the PDL file which described the contents which are all the pages that the conventional printer driver creates as a comment sentence, also by the conventional print server which cannot recognize Page ID, this PDL file can be interpreted and it can process satisfactory.

[0071] The [1st modification] With the above-mentioned operation gestalt, client equipment 10 created the PDL description showing all the pages of a document, and was embedding the page ID of each page into it. With the above-mentioned operation gestalt, since this took into consideration conventional client equipment and compatibility with the conventional print server, it was because it was premised on the ability not to know information on the page cache which the print server has managed by the client equipment side. For this reason, even when some pages in a document already existed in the page cache by the side of a print server as a page-image file, with client equipment, the PDL file of the page of ***** had to be created and, as a result, the description which will be skipped by the print server side was also included in the PDL file.

[0072] On the other hand, in this modification, by the interactive session of client equipment 10 and a print server 20, the information on the page cache 26 of a print server 20 is acquired by the client equipment 10 side, and the amount of PDL description is reduced based on this.

[0073] That is, the printer driver 14 of client equipment 10 precedes performing PDL description of each page, and asks a print server 26 whether a page image with the page ID of the page received from application 12 already exists in the page cache 26 of a print server 20. In the printer server 20 which received this, it investigates whether the file of a page image whose job management equipment 22 searches the page cache 26, and has the page ID is saved, and the information on the existence of the result, i.e., a page image, is

returned to a printer driver 14.

[0074] When an applicable page image exists [the reply] in the page cache 26, a printer driver 14 describes directions of the purport which uses the page image in a page cache instead of performing PDL description about the contents of a page of the page. This description can be performed like the above-mentioned operation gestalt using the format of a comment sentence. Specification of the page image in a page cache in this description should just use Page ID. on the other hand, when an applicable page image does not exist [the reply from a print server 20] in a page cache, a printer driver 14 boils the PDL description about the contents of the page as usual, performs it, and adds the page ID of the page concerned in the form of a comment sentence and others.

[0075] In addition, a printer driver 14 performs PDL description as usual, when it turns out that the destination of a job is a print server without the page cache 26.

[0076] The print server 20 interprets the PDL file received from client equipment 10 sequentially from the head, and when comment description of the purport which uses the specific page image in the page cache 26 is detected, it should just register into a page ID list the page ID specified within the description. Since the interactive session at the time of PDL file creation has already shown that the page image with the page ID is saved into the page cache 26, it is not necessary to search the page cache 26 with a print server 20 anew. About other processings, it is the same as that of the above-mentioned operation gestalt, and is good.

[0077] Thus, in this modification, the PDL description showing the contents of that page is omitted about the page in the page cache 26 of a print server 20 by the PDL file created by the printer driver 14. For this reason, following various effectiveness is acquired.

[0078] First, by the client equipment 10 side, since the PDL file which high processing of the load of describing a page decreases and creates becomes small, a load is reduced and improvement in the speed of processing can be expected. For example, since the image of all pages already exists in the page cache 26 when it is going to print again in the completely same condition, without changing the same document as the time of printing before, the size of the PDL file created becomes min and is the most efficient. Therefore, the application 12 and the printer driver 14 of client equipment 10 are wide opened immediately from printing processing.

[0079] Moreover, speaking of a network 40, since the size of the PDL file to transmit becomes small, reduction of a load and improvement in the speed of a file transfer can be expected.

[0080] Moreover, since the PDL file to process is small and can be managed with a print server 20 side, there is little disk capacity which is needed with job storage equipment 23, it ends, the processing load of the print server 20 whole is also reduced, and improvement in the speed can be expected as a result.

[0081] In addition, printing processing can be performed satisfactory with the print server 20 of this modification by performing the same processing as the above-mentioned operation gestalt also with the client equipment 10 of the above-mentioned operation

gestalt which cannot have an interactive session.

[0082] Thus, according to this modification, the amount of information processed by the whole system decreases, and there are few required resources, such as storage capacity in each part of a system and communication link time amount, and it ends, and a processing load becomes light, and processing is accelerated.

[0083] With the [2nd modification] above-mentioned implementation gestalt, it came by the print server 20 considering the unit of the image data which carries out a cache as a page of a document (page cache 26). However, it is necessary to limit no cache units to a page. Here, other examples of the unit of the cache of an image are explained.

[0084] First, it is possible to make a document into the unit of a cache. For example, in a document format called PDF (Portable Document Format) of Adobe Systems, since it makes it indispensable to put in unique ID of the document itself, and unique ID for every version of a document in a file, each document is uniquely discriminable for every version. These ID is used and how to carry out the cache of the whole document by the print server 20 can be considered. With the cache equipment of the image data of a print server 20, all the pages that constitute a document are managed collectively. The approach of carrying out a cache for every document by the document on condition of modification of contents like a PDF file is very effective in improvement in the speed of re-printing.

[0085] Moreover, it is also suitable to make into the unit of a cache each drawing object contained in a document. A drawing object here is the physical component of a document smaller than a page, for example, they are a letter face (text), a graphic form frame, etc. (graphics, continuous tone image, etc.).

[0086] In this modification, client equipment 10 gives ID of a proper to such each drawing object, respectively, includes it in a PDL file, in a print server 20 side, matches the image data of each created drawing object with that ID, and carries out a cache. In the image data which carries out a cache, the information on the location within the page of the drawing object concerned is matched and kept. Client equipment 10 will give new ID to the object after modification, if modification of Make Changes, migration, zooming, etc. is added to a drawing object.

[0087] When the job of re-printing of the document which corrected the part from client equipment 10 comes, in a print server 20, ID of each drawing object of this job is detected from a PDL file, it investigates whether the image data with that ID is saved to cache equipment, and when saved, that image data is read from cache equipment, and is reused. When the cache of the image data of the ID is not carried out, the usual deconstructivism pause processing is performed about PDL description of the drawing object. In this case, deconstructivism POZA 25 compounds the image generated by the image and deconstructivism pause processing which were read from cache equipment, and generates a 1-page image.

[0088] Thus, if the cache of the image data is carried out per drawing object, the image data created once can be reused more efficiently and re-printing processing can be accelerated. Moreover, if a cache is carried out in a unit smaller than a page, the use

effectiveness of a cache will also rise.

[0089] In addition, with the configuration which performs a cache per drawing object, the count of the cache retrieval at the time of printing tends to increase more than the case of a page unit, and the time amount which one retrieval takes also tends to become long. This point is improvable by taking the property of each drawing object into consideration.

[0090] For example, in the case of a PostScript file, generally about objects, such as text data and vector drawing data (graphics), deconstructivism pause processing does not take big time amount. On the other hand, since deconstructivism pause processing has much processing a load is highly applied [processing] by time amount, such as resolution conversion, color separation, and half toning, in the case of the object of a raster image, processing takes great time amount in many cases. Moreover, although text data is edited frequently and it is updated on the other hand in the site of document preparation in many cases as known well, the frequency where a raster image is edited rather than a text has many few things.

[0091] If the property of such a drawing object is taken into consideration, time amount will be taken and deconstructivism pause processing will understand that the cache of a raster image object with being changed [little] is effective. Since the time amount which deconstructivism pause processing takes is short about a text or graphics, the need for a cache is low. Therefore, the load of deconstructivism pause processing is mitigable, avoiding increase of the cache retrieval load in a print server 20 by setting only the drawing object of a raster image as the object of a cache.

[0092] With the above-mentioned operation gestalt, since the cache of an image data was performed per page, when a raster image and a text were intermingled in a page, that the text was changed also had to carry out deconstructivism pause processing of the whole page, but by this amelioration method, since a raster part can use the image data in a cache, it can be processed at a high speed.

[0093] It is more more efficient to have joined together organically and to use among two or more print servers 20, rather than it uses the [3rd modification] page cache 26 by one print server 20. Through a network 40, organic association is print server 20 which can communicate mutually, and is performed by exchanging the information on the page cache 26.

[0094] In this modification, image formation equipment 24 asks whether the page image with that page ID is kept by the page cache 26 of that server 20 to other print servers 20, when there is no page image with the target page ID in the page cache 26. If there is a reply of the purport which is keeping the page image concerned from one on a network 40 of the print servers 20, image formation equipment 24 will memorize the identifier of the print server 20, and will tell that the page image existed in deconstructivism POZA 25. Deconstructivism POZA 25 skips deconstructivism pause processing of the page according to it.

[0095] And not only the page ID illustrated in the above-mentioned operation gestalt but the information on the identifier of a print server that the page image was kept is recorded

on the configuration page file which image formation equipment 24 creates. The IOT control device 27 acquires the file of a page image using Page ID and the print server name which were shown in the configuration page file, and supplies it to IOT30.

[0096] It is reusable if the page image is kept by either of two or more print servers 20 which are contained in a printing system according to this configuration. It overlaps and it becomes unnecessary therefore, to save the same page image every print server 20. Moreover, since each print server 20 can use equally the page cache 26 of all the print servers 20, the cache capacity of each print server 20 becomes equal to the cache total amount of all the print servers 20 belonging to a system seemingly. Thus, according to this configuration, in each print server 20, the effectiveness of the increment in quick image formation and cache capacity is acquired.

[0097] In the above, the modification was explained to the gestalt list of suitable operation of this invention. Although embedded into the PDL file by making Page ID into a comment sentence above, this invention is not limited to such a method. For example, the table showing the correspondence relation between each page and Page ID can be created, and the same effectiveness can be acquired also by method which matches with a PDL file and is sent to a print server 20. Moreover, although the job storage equipment 23 which keeps the information on a job (a PDL file, a job ticket, configuration page file, etc.) was formed in the interior of a print server 20 above, a configuration which connects job storage equipment 23 to a network 40 as equipment with an another print server 20 is also within the limits of this invention. In this case, it is also possible to share one job storage equipment 23 linked to a network 40 by two or more print servers.

DESCRIPTION OF DRAWINGS

[Brief Description of the Drawings]

[Drawing 1] It is the functional block diagram showing the outline configuration of the printing system concerning this invention.

[Drawing 2] It is drawing showing an example of the PDL description in which Page ID was included.

[Drawing 3] It is the flow chart which showed the procedure of a print server focusing on processing of job management equipment.

[Drawing 4] It is drawing showing an example of the contents of the configuration page file.

[Drawing 5] It is the flow chart which shows the procedure of the image formation processing in a print server.

[Description of Notations]

10 Client equipment, 12 Application, 14 A printer driver, 20 A print server, 21 A job receiving set, 22 Job management equipment, 23 job storage equipment, 24 Image formation equipment, 25 Deconstructivism POZA, 26 A page cache, 27 An IOT control unit, 30 IOT.